FAUIA-UDCH

UNIVERSIDAD PARTICULAR DE CHICLAYO

FACULTAD DE ARQUITECTURA Y URBANISMO, INGENIERIAS y ARTE & DISEÑO

ESCUELA PROFESIONAL DE INGENIERIA INFORMATICA Y DE SISTEMAS

TESIS

“DISEÑO DE UNA INFRAESTRUCTURA DE ALTA DISPONIBILIDAD Y ADMINISTRACION DE RECURSOS UTILIZANDO SOFTWARE LIBRE Y PROPIETARIO PARA LA UNIVERSIDAD PARTICULAR DE CHICLAYO, DE OCTUBRE DE 2019 A ENERO 2020”

PARA OPTAR EL TÍTULO PROFESIONAL DE:

INGENIERO INFORMÁTICO Y DE SISTEMAS

PRESENTADO POR:

Bach. ALVITEZ ALARCON ALBERTO ADOLFO

Chiclayo, agosto de 2020

DEDICATORIA

Dedico en primer lugar y de manera especial a Dios por mantenerme en salud y bendecido por mis padres y hermanos que ellos fueron mi principal motor y motivo para mi construcción de mi vida profesional, quienes en todo momento me apoyaron en las buenas y en las malas y me guiaron en base a responsabilidades y superación del día a día, por todo ello.

Se los dedico con todo el amor y cariño que les tengo.

Alberto Alvitez Alarcón.

.

.

AGRADECIMIENTO

.

Al Ing. Oscar Castro Yoshida, por su paciencia y disponibilidad en la asesoría de mi tesis.

A cada una de las autoridades, personal administrativo y asistencial que laboran en la Universidad Particular de Chiclayo, por las facilidades brindadas para culminar con éxito el presente estudio.

DECLARACIÓN DE AUTENTICIDAD

Yo, ALVITEZ ALARCON ALBERTO ADOLFO, identificado con DNI 76186862 egresado de la Escuela de Ingeniería Informática y de Sistemas de la Facultad de Arquitectura y Urbanismo, Ingenierías y Arte & Diseño, a efectos de cumplir con las disposiciones vigentes consideradas en el reglamento de Grados y Títulos de la Universidad Particular de Chiclayo, declaro bajo juramento que el informe del trabajo de suficiencia profesional denominada:

“DISEÑO DE UNA INFRAESTRUCTURA DE ALTA DISPONIBILIDAD Y ADMINISTRACION DE RECURSOS UTILIZANDO SOFTWARE LIBRE Y PROPIETARIO PARA LA UNIVERSIDAD PARTICULAR DE CHICLAYO, DE OCTUBRE DE 2016 A ENERO 2017"

Que:

1. El informe es de mi autoría.
2. He respetado y seguido los lineamientos de diseño, reglamento y parámetros establecidos para el desarrollo del informe.
3. El informe no ha sido plagiado, pues no ha sido objeto de publicaciones, ni presentada anteriormente para obtener un grado académico previo o título profesional.
4. Los datos presentados son resultado de visitas, entrevistas y consultas y que éstos no han sido falseados, pues los resultados contribuirán a conocer la realidad problemática con respecto a la investigación.

En tal sentido asumo la responsabilidad que corresponda ante cualquier falsedad, ocultamiento u omisión tanto de los documentos como de información aportada por lo cual me someto a lo dispuesto en las normas académicas de la Universidad de Chiclayo.

Pimentel, junio de 2020

El autor

PRESENTACIÓN

Señores Miembros del Jurado:

De conformidad y cumpliendo lo estipulado en el Reglamento de Grados y Títulos de la Facultad de Arquitectura y Urbanismo, Ingenierías y Arte de la Universidad Particular de Chiclayo, para optar el Título Profesional de Ingeniero Informático y de Sistema, someto a vuestra consideración el presente informe Titulado:

“DISEÑO DE UNA INFRAESTRUCTURA DE ALTA DISPONIBILIDAD Y ADMINISTRACION DE RECURSOS UTILIZANDO SOFTWARE LIBRE Y PROPIETARIO PARA LA UNIVERSIDAD PARTICULAR DE CHICLAYO, DE OCTUBRE DE 2016 A ENERO 2017".

Gracias

ÍNDICE GENERAL

Contenido

ÍNDICE DE FIGURAS

ÍNDICE DE TABLAS

RESUMEN

El presente informe de tesis denominado "**DISEÑO DE UNA INFRAESTRUCTURA DE ALTA DISPONIBILIDAD Y ADMINISTRACION DE RECURSOS UTILIZANDO SOFTWARE LIBRE Y PROPIETARIO PARA LA UNIVERSIDAD PARTICULAR DE CHICLAYO, DE OCTUBRE DE 2016 A ENERO 2017**", está orientado a garantizar el buen funcionamiento de los sistemas informáticos con los respaldos, réplicas y redundancia de equipos de comunicaciones como de los servidores de la Universidad en mención.

Considerando la estructura brindada por la Escuela de Informática y de Sistemas de la Universidad de Chiclayo, para el desarrollo del informe de las tesis, es que presento el resumen a continuación de los capítulos:

En el Capítulo I está conformado por la descripción de la realidad problemática, la formulación del problema, hipótesis y objetivos de la presente investigación.

En el Capítulo II, detallo las bases teóricas en el cual se mencionan los elementos principales del contenido del informe, donde destacamos: los antecedentes de la investigación, el marco teórico, los términos, definiciones y descripciones de los temas a tratar en todo el proyecto.

En el Capítulo III se plantea el marco metodológico constituido por la declaración, definición y operacionalización de las variables intervinientes en esta investigación, además incluye la metodología de investigación, la población y muestra a intervenir, las técnicas e instrumentos de recolección de datos. Por último, se determina el método de análisis de datos y el aspecto ético del investigador.

En el Capítulo IV y V se establecen los resultados y la discusión de dichos resultados.

En el capítulo VI y VII se redacta las conclusiones y recomendaciones obtenidas en el desarrollo de la investigación de este informe.

En el capítulo VIII constituye la propuesta tecnológica conformada por las generalidades del proyecto, así como el análisis, diseño e implementación de la aplicación. También incluye el presupuesto y el análisis costo beneficio del proyecto.

En el capítulo IX se escribe las referencias bibliográficas revisadas desde el inicio y hasta el final del proyecto.

Por último, en el capítulo X se considera algunas explicaciones adicionales respecto a la tecnología utilizada en el presente proyecto.

Palabras Clave: Protección de datos, software de código abierto, alta disponibilidad, base de datos.

ABSTRACT

This thesis report called “DESIGN OF A HIGH AVAILABILITY AND RESOURCE MANAGEMENT INFRASTRUCTURE USING FREE AND OWNER SOFTWARE FOR THE UNIVERSITY OF CHICLAYO JANUARY - JUNE 2020", is aimed at guaranteeing the proper functioning of information systems with backups, replicas and redundancy of communications equipment such as the servers of the University in question.

The structure provided by the School of Computing and Systems of the University of Chile, for the development of the thesis report, is that I present the summary following the chapters:

In Chapter I, it consists of the description of the problematic reality, the formulation of the problem, the hypotheses and the objectives of the present investigation.

In Chapter II, I detail the theoretical bases in which the main elements of the content of the report are mentioned, highlighting the background of the research, the theoretical framework, the terms, the descriptions of the topics to be discussed throughout the project.

Chapter III presents the methodological framework constituted by the declaration, definition and operationalization of the intervening variables in this research. It also includes the research methodology, population and sample, techniques and data collection instruments. Finally, the method of analyzing the data and the ethical aspect of the researcher is determined.

In Chapter IV and V the results and the discussion of said results are resolved.

In chapters VI and VII the conclusions and recommendations are drawn up in the development of the investigation of this report.

Chapter VIII presents the technological proposal consisting of the generalities of the project, as well as the analysis, design and implementation of the application. It also includes the budget and the cost-benefit analysis of the project.

In chapter IX, the bibliographic references reviewed are written from the beginning to the end of the project.

Finally, chapter X considers some additional explanations regarding to the technology used in this project.

Key words: Data Protection, open source software, high availability, database.

CAPÍTULO I

INTRODUCCIÓN

CAPITULO I: INTRODUCCIÓN

1.1. Realidad Problemática

Muchas empresas actualmente brindan un conjunto de servicios, pero muchos de ellos fueron implementados de acuerdo como se iban presentado los requerimientos de los servicios o necesidad de acuerdo a su crecimiento de ambiente y contratación de personal, en la Universidad mientras más crecían las facultades y escuelas se iban implementando más oficinas administrativas, las que generaban más información que se necesitaba almacenar en sitios seguros y que esté disponible para las áreas requeridas en momentos oportunos, se generaban nuevos procesos que cada vez se volvían más complejos por la cantidad de alumnos con los que contaba la Universidad, y para cubrir esas necesidades cada vez más complejas es que se iban instalando nuevos equipos y servicios informáticos que cubrían las necesidades del momento sin planificaciones futuras de funcionamiento continuo, de respaldo de datos, de planificación de seguridad y crecimiento de servicios .

Actualmente no existe una disponibilidad de una aplicación informática que nos brinde información con datos críticos solicitado por los usuarios de la Universidad para la toma de decisiones o información por parte de la parte usuaria como son los alumnos o padres de familia el cual logra implicar un costo significativo para la Universidad en términos de: pérdida de productividad, ingresos económicos, alumnos y padres insatisfechos logrando una mala imagen corporativa. La información está dispersa por distintas facultades, las matriculas se realizan en cada escuela, no existe entrelazada la información de todas las facultades con la dirección de Servicios académicos en tiempo real provocando grandes malestares en los clientes de la Universidad y quedando rezagado en el mercado porque la competencias directas mantienen plataformas más integradas y complejas dirigidos a los clientes de dichas universidades, volviéndolos más atractivos para los padres de familia. La información se va almacenando en distintas computadoras personales, en mayor de los casos en las computadoras de las secretarias que serían los seudo

servidores informáticos por que ahí se almacenan todo tipo de información estando siempre en constantes peligro de vulnerabilidad contra robo de información y peligro de fallo en el hardware sin; en muchas veces, poder recuperar la información por no existir un plan de backup en los equipos personales de la Universidad.

Estas configuraciones funcionan con un bajo volumen de información, pero ya con el crecimiento de sus transacciones y aumento de los servicios y usuarios finales a los que se les brinda la información surge la necesidad de pensar en mejorar esos servicios y darle mayor seguridad y continuidad. Es por eso que nace la necesidad de realizar un estudio de la situación actual de la Universidad y pensar en mejorar las plataformas en que se brinda los servicios actuales dándole mayor seguridad y continuidad con servidores que estén disponibles las 24 horas del día el cual es el objetivo de la presente tesis.

Actualmente uno de los factores claves en el éxito de las instituciones educativas es de contar con tecnologías que respalden los procesos de sus negocios para liderar en el mercado tan competitivo y estas tecnologías deben de estar siempre disponibles y tolerantes a fallas, es decir en alta disponibilidad y así ganar la confianza de sus clientes.

.

1.2. Formulación del Problema

¿Permitirá el diseño de una infraestructura de alta disponibilidad utilizando software libre y propietario la administración de recursos para la Universidad de Chiclayo?

1.3. Hipótesis

El diseño de una infraestructura de alta disponibilidad utilizando software libre y propietario permitirá la administración de los recursos de red de la Universidad Particular de Chiclayo.

1.4. Objetivos

1.4.1. Objetivo General

Realizar el diseño de una infraestructura de alta disponibilidad y

administración de recursos utilizando software libre y propietario para la Universidad de Chiclayo.

1.4.2. Objetivos Específicos

- Analizar la situación actual de la infraestructura tecnológica referente al tipo de red LAN que usa, sistemas operativos que están implementados y planes de crecimiento.
- Analizar los servicios informáticos que se brinda actualmente en la Universidad, en que están desarrollados y en que plataforma están funcionando actualmente Windows o Linux.
- Realizar un estudio comparativo de las mejores alternativas en el mercado de las soluciones en alta disponibilidad que se adecuen con los servicios informáticos brindados para la Universidad.
- Configurar los servidores Windows Server 2012 y Linux Centos con las soluciones de Alta disponibilidad de Red, alta disponibilidad de disco y replica de Base de datos SQL Server y MySql usando software libre y propietario para luego integrarlos integrarlos realizando pruebas de continuidad ante fallos.

Capítulo II:
BASES TEÓRICAS

CAPITULO II: BASES TEÓRICAS

2.1. Antecedentes de la Investigación

a) Autor: Vargas Naula Alba Elizabeth

Fecha: 2013 – Loja - Ecuador

Nombre del proyecto:
"DISEÑO DE INFRAESTRUCTURA DE RED DE ALTA DISPONIBILIDAD PARA DATA CENTER"

Resumen:
En el proyecto se diseña una infraestructura de red que permita aumentar la disponibilidad del data center de la Universidad Técnica Particular de Loja. Para ello se levantó el levantamiento de información que permita determinar la situación actual de la red LAN que se encuentra implementada actualmente.
Se detallan las dos propuestas de solución para mejorar la disponibilidad de la infraestructura de la red y luego de realizar un análisis comparativo se elige realizar un diseño de red LAN de cable cat 6ª paralela a la red de cat 6 existentes.
Finalmente se describe los componentes de la propuesta de solución seleccionada y el presupuesto, basadas en costos referenciales del proyecto.

b) Moreno Jiménez, Manuel Alejandro
Tipán Aguas, Luis Andrés

Fecha: 2015 – Quito - Ecuador

Nombre del proyecto:
"DISEÑO DE UNA RED DE ALTA DISPONIBILIDAD PARA LA UNIVERSIDAD POLITÉCNICA SALESIANA SEDE QUITO CAMPUS SUR"

Resumen
El crecimiento de la red de datos en los próximos años con la implementación de los nuevos servicios propuestos por la Universidad Politécnica Salesiana Sede Quito Campus Sur obliga a contar con una red eficiente. Por esta razón este proyecto plantea el diseño de una red de alta disponibilidad, que consiste en una red de datos basada en la infraestructura actual para lo cual se realizará un levantamiento de la topología física y lógica para efectuar un estudio de la carga de tráfico en los horarios críticos y así obtener datos de la demanda de tráfico actual que posee la Universidad Politécnica Salesiana, con esto se realizará un dimensionamiento de la red (velocidad de transmisión, cobertura inalámbrica, calidad de servicio, etc.) por medio de una simulación. El proyecto tiene como objetivo principal el resolver los problemas de los usuarios en la red de datos y satisfacer las necesidades para el acceso a las aplicaciones y servicios que se tendrán en los próximos años en la Universidad Politécnica Salesiana Sede Quito Campus Sur.

c) CORAS BENDEZÚ, JORCH JOAN

Fecha: 2013 – Huancayo - Peru

Nombre del proyecto:

"REDISEÑO DE LA RED DE COMUNICACIONES BASADO EN TECNOLOGIAS DE ALTA DISPONIBILIDAD DE GESTIÓN DE TRÁFICO PARA MEJORAR LA COMUNICACIÓN DE LA MUNICIPALIDAD PROVINCIAL DE CHURCAMPA – HUANCAVELICA"

Resumen

El presente trabajo de Tesis Titulado: "REDISEÑO DE LA RED DE COMUNICACIONES BASADO EN TECNOLOGIAS DE ALTA DISPONIBILIDAD DE GESTIÓN DE TRÁFICO PARA MEJORAR LA COMUNICACIÓN DE LA MUNICIPALIDAD PROVINCIAL DE CHURCAMPA – HUANCAVELICA", surge debido a los problemas que ocurren en la Municipalidad Provincial de Churcampa, tales como de infraestructura, de prestación, de equipamiento, entre otros, los cuales afectan al correcto desempeño de las actividades de los trabajadores y/o empleados que laboran en dicha entidad.

Dentro del trabajo, se utilizó una población conformada por 50 empleados de la Municipalidad Provincial de Churcampa. Posteriormente, se procedió a realizar el análisis del sistema de comunicaciones.

Basado en los resultados del análisis, se presenta una propuesta de rediseño de la red de comunicaciones de la Municipalidad Provincial de Churcampa basado en tecnología de Alta Disponibilidad. Cuya implementación beneficiara a la empresa.

2.2. Marco Teórico

2.2.1. Redes de computadoras

2.2.1.1. Definición de red de computadoras

Es la interconexión física o inalámbrica que vincula varios dispositivos informáticos (servidores, computadoras, teléfonos móviles, periféricos, entre otros) para que se comuniquen entre sí, con la finalidad de compartir datos y ofrecer servicios.

Una Local Área Network (por sus siglas) o Red de Área Local, conecta equipos informáticos ubicados en un área geográfica reducida, como un edificio o una habitación.

Una Wide Área Network (por sus siglas) o Red de Área Amplia, es un conjunto de redes LAN que conecta equipos informáticos que se encuentran en diferentes ubicaciones físicas.

Red alámbrica

Es la red que conecta equipos y transmite datos a través de cables basados en el estándar Ethernet. Es ideal para el manejo de grandes cantidades de datos a velocidades muy altas, por ejemplo, en la industria de multimedia. Una red alámbrica tiene dos componentes esenciales: switches y routers.

Los switches o conmutadores permiten que los dispositivos en su red se comuniquen entre sí, recibiendo paquetes de datos y direccionándolos al destinatario correcto. Al hacer posible que la información y los recursos sean compartidos, los switches le ayudan a ahorrar dinero e incrementar la productividad.

Los enrutadores conectan múltiples redes entre sí o con Internet. Analizan los datos y los envían por la mejor ruta. Protegen la información de las amenazas de seguridad e incluso deciden qué equipos de cómputo tienen prioridad sobre otros.

Red inalámbrica (WiFi)

Es la red que permite la conexión entre dispositivos a través de ondas de radio, sin la necesidad del uso de cables. Una de sus principales ventajas es el tema de costos y la flexibilidad para soportar usuarios móviles; sin embargo, la seguridad debe ser mucho más exigente y robusta para evitar intrusos. Sus componentes esenciales son los puntos de acceso y las controladoras.

Puntos de acceso. Conectan los dispositivos con la red, sin necesidad de cables y actúan como un amplificador, extendiendo el ancho de banda para soportar más dispositivos y usuarios móviles. Los puntos de acceso proporcionan también datos útiles para mejorar el uso de la red y ofrecen seguridad proactiva.

Controladores. Su principal función es configurar y monitorear los puntos de acceso de manera práctica y sencilla, permitiéndole desarrollar y operar la red inalámbrica de forma remota, eliminando tareas repetitivas.

Características de una red

Hoy más que nunca, usted necesita una red que le ayude a ofrecer la mejor conectividad para los usuarios, de forma confiable y segura, al tiempo en que le permite obtener la información necesaria para tomar decisiones y hacer crecer su negocio. Tenga en cuenta estas características a la hora de elegir la infraestructura de red para su PyME.

Escalabilidad. Una red escalable es aquella que tiene la capacidad de reaccionar y adaptarse fácilmente al crecimiento de su negocio, de los usuarios y de las cargas de trabajo, protegiendo su inversión y asegurando la continuidad de la operación.

Seguridad. Una red segura es aquella que cuenta con las políticas y prácticas necesarias para prevenir y supervisar el acceso no autorizado, así como el uso indebido, en la información de su empresa y sus recursos. Actualmente, las amenazas a la seguridad son cada vez mayores y pueden poner en riesgo tanto la integridad como la continuidad de su negocio.

Automatización. Una red automatizada es aquella en la cual los dispositivos pueden ser configurados, aprovisionados, gestionados y probados automáticamente. Esto permite mejorar la eficiencia, evitar errores humanos y reducir los gastos operativos.

Inteligencia. Una red inteligente es capaz de extraer insights o información relevante de los dispositivos, las aplicaciones y los usuarios, para hacer más eficiente su operación y facilitar la toma de decisiones para el negocio.

Redes definidas por software

Las redes definidas por software, o SDN, son una manera de abordar la gestión de redes. En ellas, el control se desvincula totalmente del hardware y se le da a una aplicación de software llamada controlador. Esto es especialmente útil en ambientes distribuidos y de nube, porque le permite al administrador manejar cargas de tráfico de manera flexible y más eficiente.

Antes

La WAN tradicional conectaba a usuarios de sucursales o campus con aplicaciones ubicadas en servidores del centro de datos, garantizando la seguridad y la conectividad por medio de circuitos exclusivos MPLS. Esto no funciona para un mundo centrado en la nube.

Ahora

SDWAN es una WAN definida por software. Ésta permite reducir costos, simplificar la administración, reforzar la seguridad y mejorar la experiencia de los usuarios.

Soluciones Inalámbricas:

El uso de WiFi en las redes corporativas ha aumentado exponencialmente y la gestión de redes inalámbricas para una implementación aerodinámica y segura de “Bring Your Own (BYOD)” se ha convertido en un desafío. Se necesitan varios puntos de acceso inalámbricos para la cobertura de todo el sitio con el fin de manejar todos los clientes WiFi. Es posible que las PYMES no cuenten con el presupuesto para soluciones de administración centralizadas que involucren Controladores Inalámbricos. Estos, también son típicamente difíciles de integrar con la infraestructura existente en la PYMES y requieren capacitación especializada para el mantenimiento. Las Soluciones Inalámbricas de ABP ofrecen una gestión WiFi completa y centralizada, perfecta para las PYMES y las aplicaciones de los servicios de hotelería sin costos o complejidades adicionales.

2.2.2. Infraestructura de red

Hemos entrado en una era de convergencia en la nube, como nunca antes, ya que las redes, el almacenamiento y las aplicaciones se unen de manera innovadora para mejorar la eficiencia, reducir los costos y aumentar la escalabilidad. La movilidad ha sido uno de los principales facilitadores de esta convergencia. El acceso a sus archivos y aplicaciones TI desde cualquier lugar, se ha convertido en la clave para manejar todo el potencial de la nube al servicio de su negocio. De la comunicación IP a la vigilancia IP hasta el Internet de las cosas (IoT), la nube y la movilidad han impulsado aplicaciones innovadoras en todos los dominios. Todo esto, no sería posible sin una estructura robusta de infraestructura que permita unir todo.

2.2.3. Recurso de red

Los recursos de una red son todas las cosas de una computadora integrantes de la red comparten para hacer usadas por el resto, algunos ejemplos son: escáner, impresoras, archivos y otros periféricos.

2.2.4. Alta disponibilidad

Los sistemas de alta disponibilidad garantizan los servicios de forma que si el sistema cae, sea reemplazado de forma automática.

La alta disponibilidad se basa en la duplicación del hardware, de esta forma, en caso de que algo falle en un sistema, tendremos otro servidor idéntico que lo sustituirá automáticamente, garantizando así la continuidad del servicio.

Imaginemos que tenemos un Servidor en el que se almacenan los datos del programa de gestión de una empresa. Tras un fallo en su funcionamiento, es necesario sustituir un componente de dicho sistema. El componente puede tardar mínimo 24h en recibirse del proveedor en caso de que la tenga en stock, por lo tanto el tiempo que pasa hasta que se soluciona el problema es muy alto. En este momento, la empresa queda parada y no puede continuar su actividad y las pérdidas económicas generadas por este parón pueden ser considerables.

Con alta disponibilidad: highavaiable

El servidor de almacenamiento dispone de una máquina de las mismas características que está totalmente sincronizada y simplemente el sistema sigue funcionando con normalidad. El tiempo de respuesta de dicho sistema son minutos.

Sistemas de alta disponibilidad

Actualmente los sistemas de alta disponibilidad cuentan con dos tecnologías:

Alta disponibilidad Activo - Pasivo: Consta de dos servidores idénticos, uno de ellos activo ofreciendo los servicios y un segundo servidor, el pasivo que está en continua sincronización a la espera de una conmutación por error o programada.

Alta disponibilidad Activo - Activo: Consta de un servidor en el cual está todo duplicado. Este servidor es capaz de ejecutar los procesos con dos procesadores, RAM y controladoras de disco funcionando de forma simultánea y aprovechando su potencia. En caso de rotura de uno de los sistemas internos, pasaría toda la carga a uno de los procesadores.

La norma IEEE 762/2006 nos muestra la forma como se debe de calcular la disponibilidad de un servicio.

Los indicadores relacionados con la disponibilidad son al menos seis:

- Disponibilidad (propiamente dicha)
- Fiabilidad
- Tiempo medio entre paradas (se conoce a menudo como MTBF, Mid Time Between failures, aunque con este nombre haría solo referencia a paradas por fallas y no a paradas en general sea cual sea el motivo)
- Duración de las paradas (se conoce a menudo como MTTR, o Mid Time To Repair, aunque con este nombre haría solo referencia a paradas por fallas y no a paradas en general sea cual sea el motivo)
- Número de paradas por mantenimiento
- Tiempo total perdido por mantenimiento

Para su cálculo es necesario previamente elaborar una tabla con los siguientes datos, para cada ítem (planta, área, sistema, subsistema o equipo) del que se pretenda obtener resultados:

- Número de paradas registradas

- Motivo de cada parada (mantenimiento programado, mantenimiento no programado, modificación, etc.)
- Duración de cada parada, preferiblemente en minutos

Con estos datos y aplicando las fórmulas que se describen, ya pueden obtenerse los valores de cada uno de ellos para un determinado periodo de tiempo.

Disponibilidad

La disponibilidad propiamente dicha es el cociente entre el tiempo disponible para producir y el tiempo total de parada. Para calcularlo, es necesario obtener el tiempo disponible, como resta entre el tiempo total, el tiempo por paradas de mantenimiento programado y el tiempo por parada no programada. Una vez obtenido se divide el resultado entre el tiempo total del periodo considerado.

$$Disponibilidad = \frac{Horas\ Totales - Horas\ parada\ por\ mantenimiento}{Horas\ Totales}$$

Las horas de parada por mantenimiento que deben computarse son tanto las horas debidas a paradas originadas por mantenimiento programado como el no programado.

Fiabilidad

La fórmula de cálculo es muy parecida a la anterior, pero sustituyendo en el numerador las horas de parada por mantenimiento por horas de parada por mantenimiento no programado:

$$Fiabilidad = \frac{Horas\ Totales - Horas\ parada\ por\ mantenimiento\ no\ programado}{Horas\ Totales}$$

Tiempo medio entre paradas (TMEP)

Es el tiempo medio que ha transcurrido entre dos paradas de mantenimiento, y se requiere para su cálculo en el numerador las horas totales del periodo, y en denominador, el número de paradas:

$$TMEP = \frac{Horas\ Totales\ del\ periodo}{Numero\ de\ paradas}$$

Tiempo medio hasta puesta en marcha (TMPM)

Representa el tiempo medio de duración de las diversas paradas ocurridas en el periodo e ítem analizado:

$$TMPM = \frac{Horas\ Totales\ de\ parada}{Numero\ de\ paradas}$$

Número de paradas

Sin ninguna fórmula representa el número total de eventos que han provocado paradas debidos a mantenimiento, y representa un indicador en sí mismo.

Horas totales de parada

Es la suma de todas las horas de parada que ha sufrido un determinado ítem en el periodo analizado.

2.2.5. RAID.

Este sistema RAID consiste en agrupar una serie de discos duros para que trabajen de forma conjunta y compartiendo el espacio de almacenamiento común para evitar perder tus datos si uno de los discos falla o mejorar la velocidad de lectura y escritura de los discos.

La matriz en RAID 0 distribuye los datos de manera equitativa entre las unidades que lo forman, sin incluir redundancia. Esta distribución equitativa de los datos hace que, prácticamente, se duplique la velocidad de acceso dado que cada disco transmite simultáneamente los datos de forma paralela, aumentando exponencialmente su rendimiento en relación al número de unidades que forman la matriz.

RAID1, Este tipo de RAID duplica el contenido de un disco en el resto de discos que forman la matriz. De ese modo, se aumenta

exponencialmente la tolerancia a fallos. Dicho de otra forma, si uno de los discos se estropea, al haber duplicado los datos, la información se mantiene intacta en el disco que todavía funciona.

RAID 5, es otro de los tipos de RAID más comunes. Este tipo de matrices de almacenamiento consisten en un sistema de almacenamiento distribuido entre todos los discos que forman el conjunto, a los que se les añade información de paridad, ofreciendo una excelente tolerancia a los fallos. De hecho, podría continuar funcionando incluso cuando uno de los discos ha fallado.

2.2.6. Estándar ANSI/TIA-942 Telecommunications Infrastructure Standard for Data.

Concebido como una guía para los diseñadores e instaladores de centros de datos (Data Centers), el estándar TIA942 (2005) proporciona una serie de recomendaciones y directrices (guidelines) para la instalación de sus infraestructuras.

Estándar TIA942 probado en 2005 por ANSI-TIA (American National Standards Institute – Telecomunications Industry Association), clasifica a este tipo de centros en varios grupos, llamados TIER, indicando así su nivel de fiabilidad en función del nivel de disponibilidad.

Al diseñar los centros de datos conforme a la norma, se obtienen ventajas fundamentales, como son:

- Nomenclatura estándar.
- Funcionamiento a prueba de fallos.
- Aumento de la protección frente a agentes externos.
- Fiabilidad a largo plazo, mayores capacidades de expansión y escalabilidad.

De acuerdo con el estándar TIA-942, la infraestructura de soporte de un Data Center estará compuesta por cuatro subsistemas:

- Telecomunicaciones: Cableado de armarios y horizontal, accesos redundantes, cuarto de entrada, área de distribución, backbone, elementos activos y alimentación redundantes, patch panels y latiguillos, documentación.
- Arquitectura: Selección de ubicación, tipo de construcción, protección ignífuga y requerimientos NFPA 75(Sistemas de protección contra el fuego para información), barreras de vapor, techos y pisos, áreas de oficina, salas de UPS y baterías, sala de generador, control de acceso, CCTV, NOC (Network Operations Center – Centro operativo).
- Sistema eléctrico: Número de accesos, puntos de fallo, cargas críticas, redundancia de UPS y topología de UPS, puesta a tierra, EPO (Emergency Power Off- sistemas de corte de emergencia) baterías, monitorización, generadores, sistemas de transferencia.
- Sistema mecánico: Climatización, presión positiva, tuberías y drenajes, CRACs y condensadores, control de HVAC (High Ventilating Air Conditionning), detección de incendios y sprinklers, extinción por agente limpio (NFPA 2001), detección por aspiración (ASD), detección de líquidos.

Asimismo, y siguiendo las indicaciones del estándar, un CPD deberá incluir varias áreas funcionales:

- Una o varias entradas al centro.
- Área de distribución principal.
- Una o varias áreas de distribución principal.
- Áreas de distribución horizontal
- Área de equipo de distribución.
- Zona de distribución.

- Cableado horizontal y backbone.

El concepto de TIER

El nivel de fiabilidad de un centro de datos viene indicado por uno de los cuatro niveles de fiabilidad llamados TIER, en función de su redundancia (anexo G). A mayor número de TIER, mayor disponibilidad, y por tanto mayores costes de construcción y mantenimiento.

TIER	% Disponibilidad	% Parada	Tiempo anual de parada
TIER I	99,67%	0,33%	28,82 horas
TIER II	99,74%	0,25%	22,68 horas
TIER III	99, 982 %	0,02%	1,57 horas
TIER IV	100,00%	0,01%	52,56 minutos

Tabla 1:Valores TIER

Sistema de cobreTIER I- Nivel 1 (Básico)

- Disponibilidad del 99,671 %.
- Sensible a las interrupciones, planificadas o no.
- Un solo paso de corriente y distribución de aire acondicionado, sin componentes redundantes.
- Sin exigencias de piso elevado.
- Generador independiente.
- Plazo de implementación: 3 meses.
- Tiempo de inactividad anual: 28,82 horas.
- Debe cerrarse completamente para realizar mantenimiento preventivo.

TIER II- Nivel II (Componentes redundantes)

- Sistemas ópticos Disponibilidad del 99,741 %.
- Menor sensibilidad a las interrupciones.
- Un solo paso de corriente y distribución de aire acondicionado, con un componente redundante.

- Incluye piso elevado, UPS y generador.
- Plazo de implementación: 3 meses.
- Tiempo de inactividad anual: 28,82 horas.
- Plazo de implementación: 3 a 6 meses.
- Tiempo de inactividad anual: 22,0 horas.
- El mantenimiento de la alimentación y otras partes de la infraestructura requieren de un cierre de procesamiento.

TIER III- Nivel III (Mantenimiento concurrente)

- Soluciones de alta densidad Disponibilidad 99,982 %.
- Interrupciones planificadas sin interrupción de funcionamiento, pero posibilidad de problemas en las no previstas.
- Múltiples accesos de energía y refrigeración, por un solo encaminamiento activo. Incluye componentes redundantes (N+1).
- Plazo de implementación: 15 a 20 meses.
- Tiempo de inactividad anual: 1,6 horas.

TIER IV- Nivel IV (Tolerante a errores)

- 99,995 % de disponibilidad.
- Interrupciones planificadas sin interrupción de funcionamiento de los datos críticos. Posibilidad de sostener un caso de improviso sin daños críticos.
- Múltiples pasos de corriente y rutas de enfriamiento. Incluye componentes redundantes. Incluye componentes redundantes (2(N+1))- 2 UPS cada uno con redundancia (N+1).
- Plazo de implementación: 15 a 20 meses.
- Tiempo de inactividad anual: 0,4 horas.
- Novedades introducidas por la Norma 942A

2.2.7. Software libre

Software libre es el software que respeta la libertad de los usuarios y la comunidad. A grandes rasgos, significa que los usuarios tienen la libertad de ejecutar, copiar, distribuir, estudiar, modificar y mejorar el software. Es decir, el software libre es una cuestión de libertad, no de precio. Para entender el concepto, piense en «libre» como en «libre expresión», no como en «barra libre». En inglés, a veces en lugar de «free software» decimos «libre software», empleando ese adjetivo francés o español, derivado de «libertad», para mostrar que no queremos decir que el software es gratuito.

Promovemos estas libertades porque todos merecen tenerlas. Con estas libertades, los usuarios (tanto individualmente como en forma colectiva) controlan el programa y lo que este hace. Cuando los usuarios no controlan el programa, decimos que dicho programa «no es libre», o que es «privativo». Un programa que no es libre controla a los usuarios, y el programador controla el programa, con lo cual el programa resulta ser un instrumento de poder injusto.

Las cuatro libertades esenciales

Un programa es software libre si los usuarios tienen las cuatro libertades esenciales:

La libertad de ejecutar el programa como se desee, con cualquier propósito (libertad 0).

La libertad de estudiar cómo funciona el programa, y cambiarlo para que haga lo que usted quiera (libertad 1). El acceso al código fuente es una condición necesaria para ello.

La libertad de redistribuir copias para ayudar a otros (libertad 2).

La libertad de distribuir copias de sus versiones modificadas a terceros (libertad 3). Esto le permite ofrecer a toda la comunidad la oportunidad

de beneficiarse de las modificaciones. El acceso al código fuente es una condición necesaria para ello.

Un programa es software libre si otorga a los usuarios todas estas libertades de manera adecuada. De lo contrario no es libre. Existen diversos esquemas de distribución que no son libres, y si bien podemos distinguirlos en base a cuánto les falta para llegar a ser libres, nosotros los consideramos contrarios a la ética a todos por igual.

En cualquier circunstancia, estas libertades deben aplicarse a todo código que pensemos utilizar hacer que otros utilicen. Tomemos por ejemplo un programa A que automáticamente ejecuta un programa B para que realice alguna tarea. Si se tiene la intención de distribuir A tal cual, esto implica que los usuarios necesitarán B, de modo que es necesario considerar si tanto A como B son libres. No obstante, si se piensa modificar A para que no haga uso de B, solo A debe ser libre; B no es relevante en este caso.

Software libre no significa que no es comercial. Un programa libre debe estar disponible para el uso comercial, la programación comercial y la distribución comercial. La programación comercial de software libre ya no es inusual; el software libre comercial es muy importante. Puede haber pagado dinero para obtener copias de software libre, o puede haber obtenido copias sin costo. Pero sin tener en cuenta cómo obtuvo sus copias, siempre tiene la libertad de copiar y modificar el software, incluso de vender copias.

Un programa libre debe ofrecer las cuatro libertades a todo usuario que obtenga una copia del software, siempre y cuando el usuario haya respetado las condiciones de la licencia libre que cubre el software. Privar de alguna de esas libertades a ciertos usuarios, o exigirles un pago en dinero o en especie para ejercerlos, equivale a no garantizarles las libertades en cuestión, lo que hace que el programa no sea libre.

GNU es un sistema operativo de software libre, es decir, respeta la libertad de los usuarios. El sistema operativo GNU consiste en paquetes

de GNU (programas publicados específicamente por el proyecto GNU) además de software libre publicado por terceras partes. El desarrollo de GNU ha permitido que se pueda utilizar un ordenador sin software que atropelle nuestra libertad.

GNU es un sistema operativo de tipo Unix, lo cual significa que se trata de una colección de muchos programas: aplicaciones, bibliotecas, herramientas de desarrollo y hasta juegos. El desarrollo de GNU, iniciado en enero de 1984, se conoce como Proyecto GNU. Muchos de los programas de GNU se publican bajo el auspicio del Proyecto GNU y los llamamos paquetes de GNU.

El nombre GNU es un acrónimo recursivo de GNU No es Unix.

En un sistema de tipo Unix, el programa que asigna los recursos de la máquina y se comunica con el hardware se denomina núcleo. GNU se usa generalmente con un núcleo llamado Linux. Esta combinación es el sistema operativo GNU/Linux. Millones de personas usan GNU/Linux, aunque muchos lo llaman erróneamente Linux.

El desarrollo del núcleo propio de GNU, Hurd, se inició en 1990 (antes de que comenzara el de Linux). Programadores voluntarios continúan desarrollando Hurd por tratarse de un proyecto técnico interesante.

Qué es CentOS?

CentOS Linux es una distribución mantenida por la comunidad y derivada de los paquetes fuentes liberados al público por Red Hat para Red Hat Enterprise Linux (RHEL). De tal forma, CentOS Linux está enfocado en ser operacionalmente compatible con RHEL. El Proyecto CentOS principalmente cambia paquetes para eliminar las marcas comerciales y trabajos artísticos de Red Hat. La redistribución de CentOS Linux es libre y no hay que pagarlo. Cada versión de CentOS es mantenida por 10 años (por medios de actualizaciones de seguridad -- la

duración de los intervalos de mantenimiento han variado a lo largo del tiempo con relación a los paquetes fuentes liberados). Una versión nueva de CentOS es liberada aproximadamente cada 2 años y cada versión de CentOS es periódicamente actualizada (cada 6 meses) para incorporar nuevo hardware. Esto resulta en un entorno Linux seguro, de bajo mantenimiento, confiable, predecible y reproducible.

2.2.8. Software Propietario

Software no libre (también llamado software propietario, software privativo, software privado, software con propietario o software de propiedad). Se refiere a cualquier programa informático en el que los usuarios tienen limitadas las posibilidades de usarlo, modificarlo o redistribuirlo (con o sin modificaciones, o cuyo código fuente no está disponible o el acceso a éste se encuentra restringido. Para la Fundación para el Software Libre (FSF) este concepto se aplica a cualquier software que no es libre o que sólo lo es parcialmente (semilibre), sea porque su uso, redistribución o modificación está prohibida, o requiere permiso expreso del titular del software. En el software no libre una persona física o jurídica (compañía, corporación, fundación, etc.) posee los derechos de autor sobre un software negando o no otorgando, al mismo tiempo, los derechos de usar el programa con cualquier propósito; de estudiar cómo funciona el programa y adaptarlo a las propias necesidades (donde el acceso al código fuente es una condición previa); de distribuir copias; o de mejorar el programa y hacer públicas las mejoras (para esto el acceso al código fuente es un requisito previo).

Windows Server es una distribución de Microsoft para el uso de servidores. Está desarrollado en el lenguaje de programación C++ y Asembler. Se trata de un sistema multiproceso y multiusuario que a día de hoy utilizan millones de empresas de todo el mundo gracias a las características y ventajas que ofrece. La primera versión del sistema fue Windows 2000 Server, lanzada a principios del nuevo milenio. Fue

concebida para ser el servidor de archivos, impresión y web de PYMEs. Una solución extraordinaria para cuando no era necesario contar con un servidor dedicado a cada tarea, pudiendo así tener todo centralizado en un único servidor. Era capaz de soportar hasta cuatro procesadores. A lo largo del tiempo, Microsoft, poco a poco ha añadido mejoras notables en este sistema Windows Server 2019.

2.2.9. **Base de Datos**

2.2.9.1. Concepto

Es un contenedor que permite almacenar la información de forma ordenada con diferentes propósitos y usos. El propósito de las bases de datos surge con la necesidad de registrar y almacenar datos. Por muchos años la mejor forma de hacer esto consistía en un archivo de documentos en papel, pero pronto esto se volvió ineficiente, pues lo siguiente después de guardar datos es poderlos consultar fácilmente.

2.2.9.2. Tipos de Base de Datos

Las bases de datos de tipo OLTP (On Line Transaction Processing) también son llamadas bases de datos dinámicas lo que significa que la información se modifica en tiempo real, es decir, se insertan, se eliminan, se modifican y se consultan datos en línea durante la operación del sistema. Un ejemplo es el sistema de un supermercado donde se van registrando cada uno de los artículos que el cliente está comprando y a su vez el sistema va actualizando el Inventario.

Las bases de datos de tipo OLAP (On Line Analytical Processing) también son llamadas bases de datos estáticas lo que significa que la información en tiempo real no es afectada, es decir, no se insertan, no se eliminan y tampoco se modifican datos; solo se realizan consultas sobre los datos ya existentes para el análisis y toma de decisiones. Este tipo de

bases de datos son implementadas en Business Intelligence para mejorar el desempeño de las consultas con grandes volúmenes de información.

2.2.9.3. Modelos de Base de Datos

Existen diferentes maneras de ordenar y organizar la información para que este sea accesible para nosotros. No existe el sistema de base de datos perfecto: hay que elegir aquella estructura que mejor se adapte a nuestras necesidades. Los siguientes son los tipos más comunes:

- Las bases de datos jerárquicas construyen una estructura de jerarquía con los datos que permite una estructuración muy estable cuando gestionamos una gran cantidad de datos muy interrelacionados.
- Las bases de datos en red derivan de las jerárquicas, pero mejoran la gestión de datos redundantes manteniendo su rendimiento en consultas de datos.
- Las bases de datos transaccionales están diseñadas para el envío y recepción de datos a grandes velocidades y de forma continua. Su único fin es la recepción y envío de información, pero la gestión de almacenamiento o redundancia están fuera de su propósito.
- Las bases de datos relacionales son las más utilizadas en aplicaciones reales. La información se almacena siempre haciendo referencia a otra por lo que se facilita la gestión y su uso por personal no especialista. En este modelo el lugar y la forma donde se guarde la información es secundario.
- Las bases de datos orientadas a objetos han surgido como concepto tras la aparición de los sistemas de programación orientada a objetos.

- Las bases de datos documentales están especializadas en el almacenamiento de textos completos, por lo que facilitan el tratamiento informatizado de grandes cadenas de caracteres.

2.2.9.4. Sistema de gestión de bases de datos relacionales (RDBMS).

Un sistema de gestión de bases de datos relacionales (RDBMS) es un programa que te permite crear, actualizar y administrar una base de datos relacional. La mayoría de los RDBMS comerciales utilizan el lenguaje de consultas estructuradas (SQL) para acceder a la base de datos, aunque SQL fue inventado después del desarrollo del modelo relacional y no es necesario para su uso.

Los principales productos RDBMS son Oracle, DB2 de IBM y Microsoft SQL Server. A pesar de los desafíos repetidos por tecnologías de la competencia, así como la afirmación de algunos expertos que dicen que ninguno de los RDBMS actuales ha aplicado plenamente los principios relacionales, la mayoría de las nuevas bases de datos corporativas siguen siendo creadas y gestionadas con un RDBMS (IBM, Base de datos, 2014).

SQL Server.

Microsoft SQL Server es un sistema de gestión de base de datos relacional, desarrollado por la empresa Microsoft. El lenguaje de desarrollo utilizado (por línea de comandos o mediante la interfaz gráfica de Management Studio) es Transact-SQL (TSQL), una implementación del estándar ANSI del lenguaje SQL, utilizado para manipular y recuperar datos (DML), crear tablas y definir relaciones entre ellas (DDL).
Entre sus características más importantes tenemos:

- Soporte de transacciones.
- Soporta procedimientos almacenados.
- Incluye también un entorno gráfico de administración, que permite el uso de comandos DDL y DML gráficamente.
- Permite trabajar en modo cliente-servidor, donde la información y datos se alojan en el servidor y los terminales o clientes de la red sólo acceden a la información.
- Además, permite administrar información de otros servidores de datos.

Oracle.

Oracle es un sistema de gestión de base de datos relacional (Relational Data Base Management System), desarrollado por Oracle Corporation.

Se considera a oracle como uno de los sistemas de bases de datos más completos, destacando:

- Soporte de transacciones
- Estabilidad
- Escalabilidad
- Soporte Multiplataforma

Aunque su dominio en el mercado de servidores empresariales ha sido casi total hasta hace poco, recientemente sufre la competencia del Microsoft SQL Server de Microsoft y de la oferta de otros RDBMS con licencia libre como PostgreSQL, MySQl o Firebird. Las últimas versiones de Oracle han sido certificadas para poder trabajar bajo GNU/linux. (Issac, Octubre 2015)

DB2.

DB2 es una familia de productos de sistema de gestión de bases de datos relacionales (RDBMS) de IBM que sirven a

varias plataformas diferentes de sistemas operativos. Según IBM, DB2 lidera en términos de participación y rendimiento en el mercado de bases de datos. Aunque los productos DB2 se ofrecen para sistemas basados en UNIX y sistemas operativos de computadoras personales, DB2 sigue a productos de base de datos de Oracle en sistemas basados en UNIX y a Access de Microsoft en sistemas Windows.

Además de su oferta para el OS/390 para mainframe y los sistemas operativos VM y sus sistemas AS/400 de su gama media, IBM ofrece productos DB2 para un espectro multiplataforma que incluye Linux basado en UNIX, HP-UX, Sun Solaris y SCO UnixWare; y para su sistema operativo de computadoras personales OS/2, así como para los sistemas Windows 2000 de Microsoft y versiones anteriores. Las bases de datos DB2 se puede acceder desde cualquier programa de aplicación mediante el uso de la interfaz Open Database Connectivity (ODBC) de Microsoft, la interfaz Java Database Connectivity (JDBC) o un corredor de interfaz CORBA. (Rouse, 2015)

Mysql.

En cuanto a la definición general, MySQL es un sistema de gestión de bases de datos relacionales de código abierto (RDBMS, por sus siglas en inglés) con un modelo cliente-servidor. RDBMS es un software o servicio utilizado para crear y administrar bases de datos basadas en un modelo relacional. Ahora, echemos un vistazo más de cerca a cada término:

Código abierto significa que eres libre de usarlo y modificarlo. Cualquiera puede instalar el software. También puedes aprender y personalizar el código fuente para que se adapte

mejor a tus necesidades. Sin embargo, la GPL (licencia pública de GNU) determina lo que puedes hacer según las condiciones. La versión con licencia comercial está disponible si necesitas una propiedad más flexible y un soporte avanzado.

Las computadoras que tienen instalado y ejecutan el software RDBMS se llaman clientes. Siempre que necesitan acceder a los datos, se conectan al servidor RDBMS. Esa es la parte "cliente-servidor". (Gustavo, 2019)

Postgresql.

PostgreSQL es un sistema gestor de bases de datos relacionales, está orientado a objetos, es multiplataforma y open source.

Está desarrollado desde 1996 por la comunidad partir del SGBD POSGRES, que surgió a partir de un proyecto de investigación militar estadounidense con participación civil.

Como ya hemos dicho es orientado a objetos, es decir, todos los elementos de nuestra base de datos van a poder tratarse como objetos, algo parecido a un lenguaje de programación.

Es multisistema, por tanto PostgreSQL puede ser instalado en Microsoft Windows, GNU/Linux, MacOS, BSD y muchos otros sistemas operativos.

Es extensible, podemos añadir funcionalidades que no vengan provistas de serie. Si quieres saber más sobre ello tenemos un artículo detallado de cómo instalar extensiones en PostgreSQL.

PostgreSQL es escalable y puede manejar bases de datos enormes, de más de 100 Terabytes y funciona bajo licencia libre, podemos usarlo para cualquier propósito sin ningún problema.

PostgreSQL se utiliza en diversos ámbitos, aquí te dejamos

algunos de los ejemplos donde se hace uso de este gestor de bases de datos:

- Almacenamiento de datos (DWH).
- En servicios como Amazon Web Services Redshift.
- Para procesado de datos, almacenado tanto en la propia instancia como en otros servicios que puedan conectarse.
- En sistemas de información geográfica, como el servicio de mapas web o también en servicios móviles OpenStreetMap.
- En bases de datos para servicios web.
- En CMS como Drupal o WordPress.
- En la conocida base de datos de cine IMDb.

También se puede usar en una plataforma como servicio: Algunos ejemplos que lo utilizan, por ejemplo Skype, que lo usa para dar servicio a todos sus departamentos y diferentes aplicativos. Y muchas otras empresas como por ejemplo Telefónica, BBVA, Petrobras, el metro de Sao Paulo o el servicio de información meteorológica del Reino Unido. (Gil, 2018).

.

2.3. Definiciónde términos

A

Alta Disponibilidad: (High availability) es un protocolo de diseño del sistema y su implementación asociada que asegura un cierto grado absoluto de continuidad operacional durante un período de medición dado.

Alta Disponibilidad de Base de Datos: Asegura la continuidad de los sistemas de base de datos mediante replicas.

B

Bases de datos: Conjunto de datos pertenecientes a un mismo contexto y almacenados sistemáticamente para su posterior uso.

C

Configuración base de datos: Es un conjunto de procedimientos o pasos utilizados y estandarizados para lograr poner en óptimas condiciones una base de datos garantizando su continuidad en el tiempo...

Contingencia Informática: Posibilidad de que algo suceda. Lo que puede o no suceder. Dentro de la seguridad informática se denomina plan de contingencia (también de recuperación de desastres o de continuación de negocios), a la definición de acciones a realizar, recursos a utilizar y personal a emplear en caso de producirse un acontecimiento intencionado o accidental que inutilice o degrade los recursos informáticos o de transmisión de datos de una organización. Es decir, es la determinación precisa del quién, qué, cómo, cuándo y dónde en caso de producirse una anomalía en el sistema de información.

D

Data Center: Un Datacenter es una construcción de un tamaño mayor en el cual se depositan equipos electrónicos necesarios para poder mantener una red de computadores, es decir, contra con la energía apropiada, con la ventilación ideal y un óptimo sistema de seguridad. Este sistema funciona bajo la modalidad de housing, esto quiere decir, prestando

alojamiento web a empresas de mayor tamaño, resguardando y recopilando su información digital.

Disponibilidad: El objetivo de la disponibilidad es garantizar el acceso a un servicio o a los recursos.

E

Ethernet: Técnicamente hablando el Institute of Electrical and Electronics Engineers (IEEE) define Ethernet como el protocolo 802.3. Por supuesto, «Ethernet» es un término mucho más sencillo de decir y posiblemente lo hayas visto ya decenas de veces por todo internet. Aparte de la terminología específica, Ethernet se refiere simplemente al tipo más común de red de área local (LAN) utilizada hoy en día

H

Host: Computadora en una red. Antes se denominaba con el término "nodo" que se utiliza en el lenguaje de definición de documentos. Muchas veces se usa como sinónimo de servidor.

I

Integridad: La verificación de la integridad de los datos consiste en determinar si se han alterado los datos durante la transmisión (accidental o intencionalmente).

K

Kerberos: Kerberos es un protocolo de seguridad creado por MIT que usa una criptografía de claves simétricas para validar usuarios con los servicios de red (evitando así tener que enviar contraseñas a través de la red). Al validar los usuarios para los servicios de la red por medio de Kerberos,

se frustran los intentos de usuarios no autorizados que intentan interceptar contraseñas en la red.

L

Linux: Los principales programas responsables de interactuar con el kernel fueron creados por la fundación GNU. Por este motivo es más correcto referirse al sistema operativo como GNU/Linux en lugar de Linux. Una distribución no es más que el conjunto del núcleo, programas de sistema y aplicaciones reunidos en un solo CD-ROM (o cualquier otro tipo de medios de comunicación). Hoy en día tenemos miles de aplicaciones para la plataforma GNU/Linux, donde cada empresa responsable de una distro elige las aplicaciones que en ella deben ser incluidas.

R

RAID : Sistema de almacenamiento de datos que utiliza múltiples unidades (discos duros o SSD), entre las cuales se distribuyen o replican los datos.

Recurso: Ayuda o medio al que se puede recurrir para conseguir un fin o satisfacer una necesidad: siempre tiene algún recurso ingenioso para salir con buen pie de las situaciones complicadas; el agua es un recurso escaso y fundamental para la vida

Red: Sistema de elementos interrelacionados que se conectan mediante un vínculo dedicado o conmutado para proporcionar una comunicación local o remota (de voz, vídeo, datos, entre otros) y facilitar el intercambio de información entre usuarios con intereses comunes.

Red local: Una red local es la interconexión de varios ordenadores y periféricos. Su extensión está limitada físicamente a un edificio o a un entorno de 200 metros.

Router: Dispositivo que une entre sí dos redes, de forma que la información que no va dirigida a la otra red, no pasa a ella.

S

Servidor: En una red, es un ordenador que proporciona servicios a otros equipos (estaciones).

Sistema operativo: Capa de enlace entre Hardware/Software, que permite a los programas abstraerse del soporte físico y verlo como una serie de recursos con estructura similar a la de los programas. Ejemplos: Linux, Unix, Windows, OS, etc.

T

Tecnologías de Información: Aquellas herramientas y métodos empleados para recabar, retener, manipular o distribuir información.

TIER : inventado por el Uptime Institute para clasificar la fiabilidad de los CPDs. Esta clasificación, forma parte del estándar ANSI/TIA-942 Telecommunications Infrastructure Standard for Data Centers

.

U

UNIX: Sistema operativo multitarea y multiusuario de gran importancia en el desarrollo y evolución de Internet.

Usuario: Persona que utiliza o trabaja con algún objeto o que es destinataria de algún servicio público, privado, empresarial o profesional.

Capítulo III:
MARCO METODOLÓGICO

CAPÍTULO III: MARCO METODOLÓGICO

3.1. Variables.

a) Variable Independiente: software libre y propietario

b) Variable dependiente: infraestructura de alta disponibilidad y administración de recursos.

3.2. Operacionalización de las variables:

Variable	Dimension	Indicadores
infraestructura de alta disponibilidad y administración de recursos	Continuidad Disponibilidad de Recursos	• Eficiencia del servicio. • Calidad de servicio. • Confiabilidad del servicio. • Visitas a servicios • Disponibilidad del servicio • Tiempo de Respuesta ante caídas del sistema
Software libre y propietario	Integración Escalabilidad	• Aumento del número de servidores. • Capacidad para hacer frente al incremento de volúmenes de trabajo..

Tabla 2: Operacionalizacion de las variables

Fuente: propia autoría

3.3. Metodología

3.3.1. Tipo de estudio:

Investigación aplicada

3.3.2. Diseño de la investigación

- Definir las variable dependiente e independiente
- Formular el problema de investigación (pregunta de investigación)
- Definir la hipótesis
- Plantear los objetivos
- Redactar el título
- Mencionar el marco metodológico.

3.4. Población, muestra y muestreo.

a. Población:

Personas que laboran en la Oficina de Informática.

b. Muestra:

03 Trabajadores: Jefe de Informática, 01 soporte y personal de apoyo

C. Muestreo

Muestreo no probalístico

3.5. Técnicas e instrumentos de recolección de datos.

Análisis documental. Se emplearán las fichas textuales y de resumen, considerando como fuentes los informes y documentos.

3.6. Encuesta.

Se empleará un cuestionario, que será aplicada a los actores involucrados, es decir al personal de la oficina de Informática.

Se utilizó un cuestionario de 05 preguntas con el cual nos permitió determinar el conocimiento y la confianza de los trabajadores involucrados en el presente proyecto.

3.7. Aspectos éticos

Se tuvo en cuenta la honestidad en los resultados de la encuesta, así como el desarrollo del proyecto considerando el contexto actual de la empresa.

Capítulo IV:
RESULTADOS

CAPITULO IV: RESULTADOS

Según Fuentes en primera instancia, todo sistema debe tener establecido un Acuerdo de Nivel de Servicio (Service Level Agreement – SLA) que defina cuánto tiempo y en qué horarios debe estar en línea. En IT se nos suele pedir que desarrollemos y cumplamos acuerdos de nivel de servicio (SLAs). Si no se analizan los puntos de fallo de un sistema, y después se calcula la disponibilidad del sistema, el SLA es defectuoso desde el principio. Para complicar aún más las cosas, diferentes personas tienen diferentes definiciones de disponibilidad. Para sistemas con mayor criticidad se tienen niveles de servicio que alcanzan las veinticuatro horas al día, de todo el año.

Normalmente se expresa la disponibilidad como un porcentaje. En ocasiones, la gente hace referencia a los "cuatro nueves" (99.99%) o los "cinco nueves" (99.999%).

Disponibilidad (%)	Tiempo offline/año	Tiempo offline/mes	Tiempo offline/día
90%	36.5 días	73 hrs	2.4 hrs
95%	18.3 días	36.5 hrs	1.2 hrs
98%	7.3 días	14.6 hrs	28.8 min
99%	3.7 días	7.3 hrs	14.4 min
99.5%	1.8 días	3.66 hrs	7.22 min
99.9%	8.8 hrs	43.8 min	1.46 min
99.95%	4.4 hrs	21.9 min	43.8 s
99.99%	52.6 min	4.4 min	8.6 s
99.999%	5.26 min	26.3 s	0.86 s
99.9999%	31.5 s	2.62 s	0.08 s

Tabla 3: Disponibilidad permitida

Ahora la gran pregunta es que hacer para lograr la disponibilidad de los servidores y/o componentes. Una alternativa es duplicar los componentes críticos y la segunda es duplicar y ver el dispositivo como un todo. En este proyecto abordamos las dos alternativas.

A continuación, mostramos un cuadro de tiempo disponible promedio viéndolos como componentes o servicios individuales:

COMPONENTE O SERVICIO BRINDADO	DISPONIBILIDAD
Servicio web	85%
Servicio de Base de datos	80%
DNS	95%
Servicio de Archivos	90%

Tabla 4: Tiempo promedio de disponibilidad de los servicios

Fuente: encuesta al personal de Informática

Esta disponibilidad está basada en una encuesta realizada al personal de informática y según sus experiencias en el campo y bitácoras de sucesos se logró formar este resultado.

Disponibilidad del componente 1: Ac1

Disponibilidad del componente 2: Ac2

Disponibilidad del componente 3: Ac3

Disponibilidad del componente N: Acné

Sistema de disponibilidad: As

Y dicho esto, estamos listos para nuestra fórmula. Cuando un sistema está constituido por N componentes que son puntos únicos de fallo, la disponibilidad del sistema se puede calcular como:

Ecuación: $As=Ac1*Ac2*Ac3*...Acn$

Reemplazando valores tenemos:

As= 85% * 80% * 95% * 90% = 58.14%

De la formula tenemos que estaría disponible si lo separamos por componentes una disponibilidad del 58.14% el servidor que sería un valor muy bajo y crítico.

Pero podemos mejorarlo implementando servicios redundantes para no tener puntos inicios críticos de fallas, que discutiremos más adelante, ya que estos valores son pre implementación.

Ahora para tener los datos correctos para lograr el cálculo de los tiempos de caída de los servicios actualmente se logró recopilar los siguientes datos del equipo de trabajo de la oficina de informática de la Universidad, como se muestra en el siguiente cuadro:

CAIDAS PREVIA IMPLEMENTACION	
FACTORES QUE AFECTAN LA CONTINUIDAD DEL SERVICIO	**HORA/AÑO**
1 Actualizaciones del software no planeado	12
2 Ataques al sistema	24
3 Falla en el disco	72
4 Falla en tarjeta de red	72
5 Falla en la placa	24
6 Falla en réplicas de datos	12
7 Caída de las base de datos	24
8 Caída de aplicaciones	12
9 Falla de fuentes de Poder	24
TOTAL	**276**

Tabla 5: Número de horas de caídas de los servicios

Recordando que los datos son tomados en referencia de 365 días al año y 24 horas al día, de los cuales nos da el número de horas al año:

Horas/año = 365 días * 24 horas/día = 8,760 horas al año

Entonces con los datos mostrados en la tabla anterior podemos detectar dos componentes de mayor factor crítico de fallas, que nos genera mayor cuello de botella ante la caída del servicio. Y estos dos puntos serán los tocados en el proyecto como réplica de componentes como son el ítem 3 que es falla en el disco e ítem 4 que es falla en la tarjeta de red. Para prevenir el ítem 3 se implementó raid 1 para discos espejos y para ítem 4 se creó balanceo de tarjetas de red en los servidores.

La segunda opción es ver a los servidores ya no como componentes si no como un todo y con los datos obtenidos en la encuesta al personal de oficina de Informática podeos formar nuestra formula:

$$Disponibilidad = \frac{(A - B)}{A} X\ 100\%$$

Donde:

$A = Horas\ comprometidas\ de\ disponibilidad$

$B = Número\ de\ horas\ \ fuera\ de\ línea$

El número de horas fuera de línea son horas de “caída del sistema” durante el tiempo de disponibilidad comprometido, es decir caídas de mantenimiento no planeado.

Ahora calculamos la disponibilidad reemplazando los valores en la formula antes mencionada, como se muestra a continuación:

$Disponibilidad$ = (8760 − 276) /8760 X 100 %

$Disponibilidad$ = 96,849315 %

Este es el valor de disponibilidad previo a la implementación de la solución propuesta. Comparando este este valor obtenido (96,84%) con un referente de páginas eléctricas, podríamos obtener el tiempo de indisponibilidad permitidos por años, meses y días para un servidor, para ello empleamos la siguiente tabla:

Disponibilidad (%)	Tiempo offline/año	Tiempo offline/mes	Tiempo offline/día
90%	36.5 días	73 hrs	2.4 hrs
95%	18.3 días	36.5 hrs	1.2 hrs
98%	7.3 días	14.6 hrs	28.8 min
99%	3.7 días	7.3 hrs	14.4 min
99.5%	1.8 días	3.66 hrs	7.22 min
99.9%	8.8 hrs	43.8 min	1.46 min
99.95%	4.4 hrs	21.9 min	43.8 s
99.99%	52.6 min	4.4 min	8.6 s
99.999%	5.26 min	26.3 s	0.86 s
99.9999%	31.5 s	2.62 s	0.08 s

Tabla 6: Cuadro de tiempo permitido de indisponibilidad.

Fuente: Páginas eléctricas

Ahora estos datos antes de la implementación debemos de compáralos con el estándar TIERs. El concepto de Tier nos indica el nivel de fiabilidad de un centro de datos asociados a cuatro niveles de disponibilidad definidos. A mayor número en el Tier, mayor disponibilidad, y por lo tanto mayores costes asociados en su construcción y más tiempo para hacerlo.

- Tier I: Centro de datos Básico: Disponibilidad del 99.671%.
- Tier II: Centro de datos Redundante: Disponibilidad del 99.741%.
- Tier III: Centro de datos Concurrentemente Mantenibles: Disponibilidad del 99.982%.
- Tier IV: Centro de datos Tolerante a fallos: Disponibilidad del 99.995%.

	Nivel de diponibilidad actual con el permitido	
	estándar ANSI/TIA-942 Telecommunications Infrastructure Standard for Data Centers	**Pre implementación de nuestros servidores**
% disponibilidad	99.671%	96.840%
horas de indisponibilidad	0	276 horas

Tabla 7: comparación Pre implementación de servidores con Tiers

De cual podemos observar que no llegamos al nivel mínimo requerido para los estándares de servidores, así que debemos de hacer la revisión e implementación de nuevas soluciones tecnológicas y es el corazón de este proyecto esta realización y mejora.

También se realizó una encuesta al personal de la Oficina de Informática sobre como calificaban el funcionamiento de los servidores durante el año anterior:

Calificación del Personal de Informática

	Calificación escala 1-10
Jefe	6
Asistente	4
Asistente	4

Tabla 8: Calificación del personal de informática pre implementación

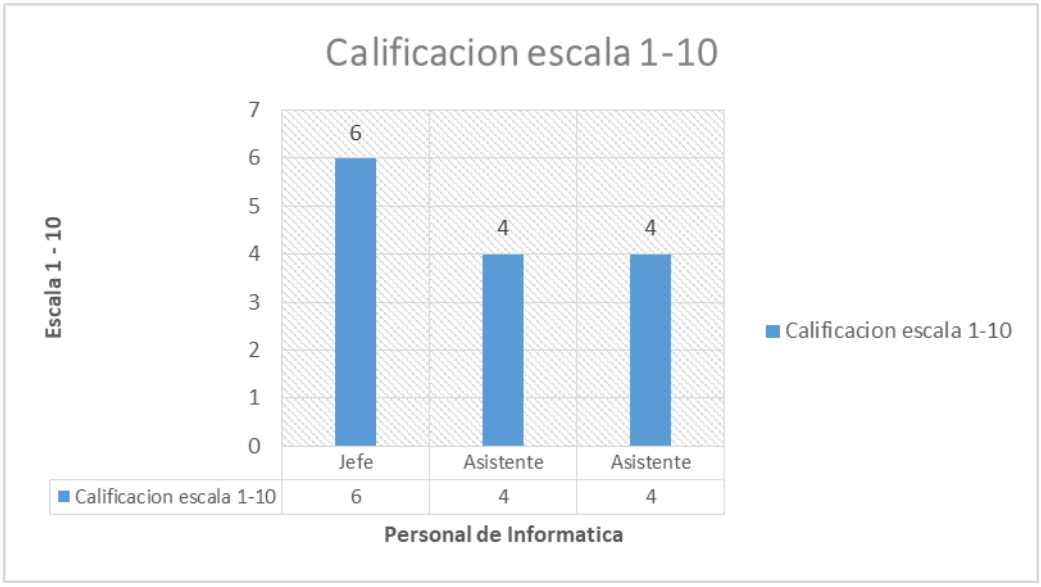

Ilustración 1: Calificación del personal de informática pre implementación

Capítulo V:
DISCUSIÓN

CAPITULO V: DISCUSIÓN

Dando una solución a la parte de ver el servidor por componentes podemos mejorar de la siguiente forma, mejorando o duplicando los servicios para no tener un solo punto de falla:

Ecuación: $As=Ac1+((1–Ac1)*Ac1)$

Usando nuestro cuadro Tiempo promedio de disponibilidad de los servicios podemos formar lo siguiente:

COMPONENTE O SERVICIO BRINDADO	FORMULA	NUEVA DISPONIBILIDAD
Servicio web	85%+((1-85%)*85%)	97.75%
Servicio de Base de datos	80%+((1-80%)*80%)	96.00%
DNS	95%+((1-95%)*95%)	99.75%
Servicio de Archivos	90%+((1-90%)*90%)	99.00%

Tabla 9: Aplicando formula de disponibilidad por componente

Del resultado anterior podemos ver que existe una mejora notable de la funcionalidad de los servicios que son componentes críticos o puntos de falla únicos cuando los duplicamos:

COMPONENTE O SERVICIO BRINDADO	ACTUAL DISPONIBILIDAD	NUEVA DISPONIBILIDAD
Servicio web	85%	97.75%
Servicio de Base de datos	80%	96.00%
DNS	95%	99.75%
Servicio de Archivos	90%	99.00%

Tabla 10: Nueva disponibilidad por componente

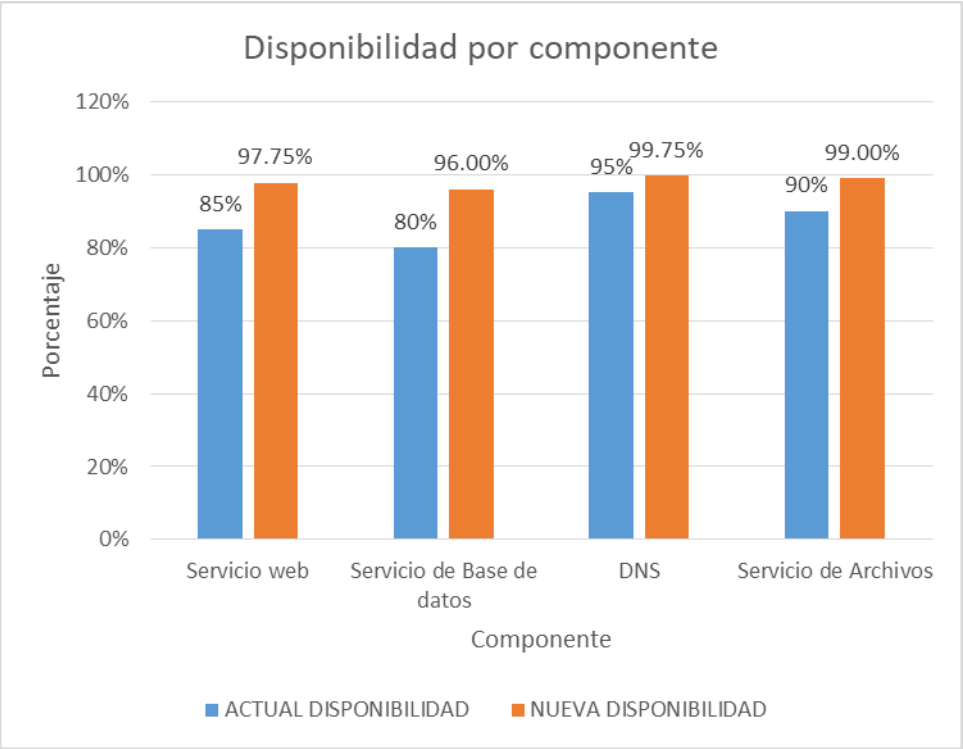

Ilustración 2: Comparativo de la duplicidad de componentes

Ahora aplicando nuevamente la fórmula para ver el total de la disponibilidad de nuestro servicio:

As= 97.75% * 96.00% * 99.75% * 99.00% =92.6693%

De la formula anterior podemos ver que el de disponibilidad de los servicios por componente usando componentes redundantes o duplicados mejora de 58.14% al 92.6693%, aunque no es lo ideal pero se podría mejorar amentando el equipamiento físico o de hardware por lo que queda demostrado que hay mejoras sustanciales al duplicar servicios.

Ahora a continuación hallaremos la mejora de disponibilidad por horas /año, nos dedicaremos a ver, según la encuesta y pruebas realizadas en nuestros equipos de prueba los resultados de nuestros servidores, según se detalla en el cuadro siguiente:

CAIDAS PREVIA IMPLEMENTACION		
	FACTORES QUE AFECTAN LA CONTINUIDAD DEL SERVICIO	**HORA/AÑO**
1	Actualizaciones del software no planeado	3
2	Ataques al sistema	8
3	Falla en el disco	0
4	Falla en tarjeta de red	0
5	Falla en la placa	1
6	Falla en réplicas de datos	2
7	Caída de las base de datos	1
8	Caída de aplicaciones	0
9	Falla de fuentes de Poder	8
	TOTAL	**23**

Tabla 11: Número de horas de caídas de los servicios post instalacion

Entonces con los datos mostrados en la tabla anterior podemos detectar que los dos componentes de mayor factor crítico de fallas desaparecen ya que existe duplicidad de los componentes.

$$Disponibilidad = (8760 - 23)\ /8760\ X\ 100\ \%$$

$$Disponibilidad = 99{,}7344\ \%$$

Este es el valor de disponibilidad post a la implementación de la solución propuesta. Comparando este este valor obtenido (99,7344%) con un referente de Tiers eléctricas, podríamos obtener el tiempo de indisponibilidad permitidos por años, meses y días para un servidor, para ello empleamos la siguiente tabla:

Disponibilidad (%)	Tiempo offline/año	Tiempo offline/mes	Tiempo offline/día
90%	36.5 días	73 hrs	2.4 hrs
95%	18.3 días	36.5 hrs	1.2 hrs
98%	7.3 días	14.6 hrs	28.8 min
99%	3.7 días	7.3 hrs	14.4 min
99.5%	1.8 días	3.66 hrs	7.22 min
99.9%	8.8 hrs	43.8 min	1.46 min
99.95%	4.4 hrs	21.9 min	43.8 s
99.99%	52.6 min	4.4 min	8.6 s
99.999%	5.26 min	26.3 s	0.86 s
99.9999%	31.5 s	2.62 s	0.08 s

Tabla 12: Cuadro de tiempo permitido de indisponibilidad.

Ahora estos datos post de la implementación debemos de compararlos con el estándar TIERs.

Tier I: Centro de datos Básico: Disponibilidad del 99.671%.

Tier II: Centro de datos Redundante: Disponibilidad del 99.741%.

Tier III: Centro de datos Concurrentemente Mantenibles: Disponibilidad del 99.982%.

Tier IV: Centro de datos Tolerante a fallos: Disponibilidad del 99.995%.

	Nivel de diponibilidad actual con el permitido	
	estándar ANSI/TIA-942 Telecommunications Infrastructure Standard for Data Centers	**Post implementación de nuestros servidores**
% disponibilidad	99.671%	99.7344%
horas de indisponibilidad	0	23 horas

Tabla 13: Nivel de disponibilidad actual (Post implementación) con el permitido

Del cuadro anterior podemos de interpretar que con nuestra implementación mejoraremos notablemente el servicio de los servidores, llegando a lograr alcanzar el nivel básico Tiers: Centro de datos Básico: Disponibilidad del 99.671%.

Si tenemos en cuenta los parámetros del Estándar ANSI/TIA-942 podemos afirmar que, al implementar una infraestructura tecnológica de alta disponibilidad, se mejora la disponibilidad de los servidores, con un nivel de disponibilidad del 99,7344 y este resultado es cercano a un Tier II (99,741%), que garantiza un “Centro de Datos Redundante” porque se llega a un estándar de calidad para Data Center que garantiza alta disponibilidad, algo que no sucedía en el estado previa implementación.

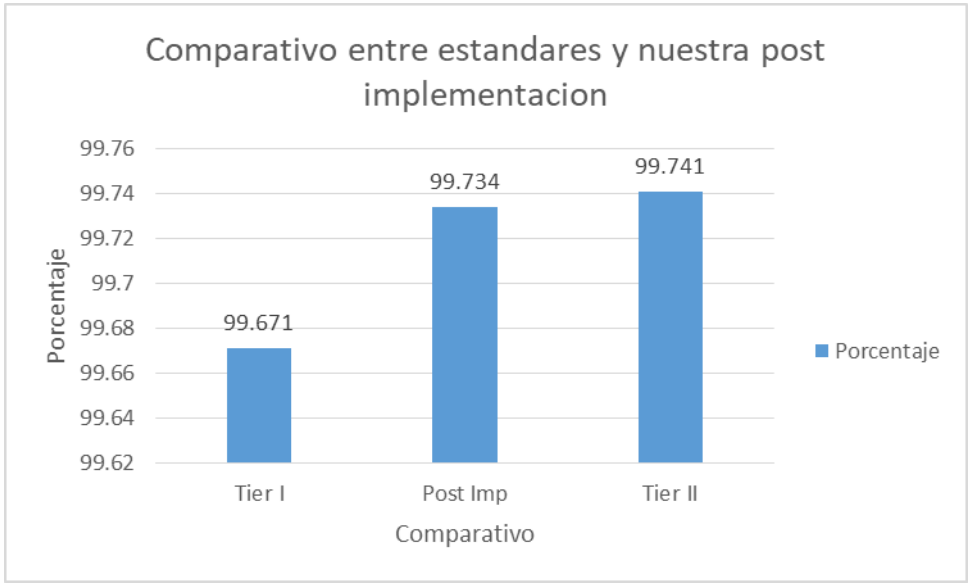

Ilustración 3: Comparativo entre estándares y nuestra post implementación

También se realizó una encuesta al personal de la Oficina de Informática sobre como calificaban el funcionamiento de los servidores post implementación:

Calificación del Personal de Informática

	Calificación escala 1-10
Jefe	9
Asistente	7
Asistente	8

Tabla 14: Calificación del personal de informática pre implementación

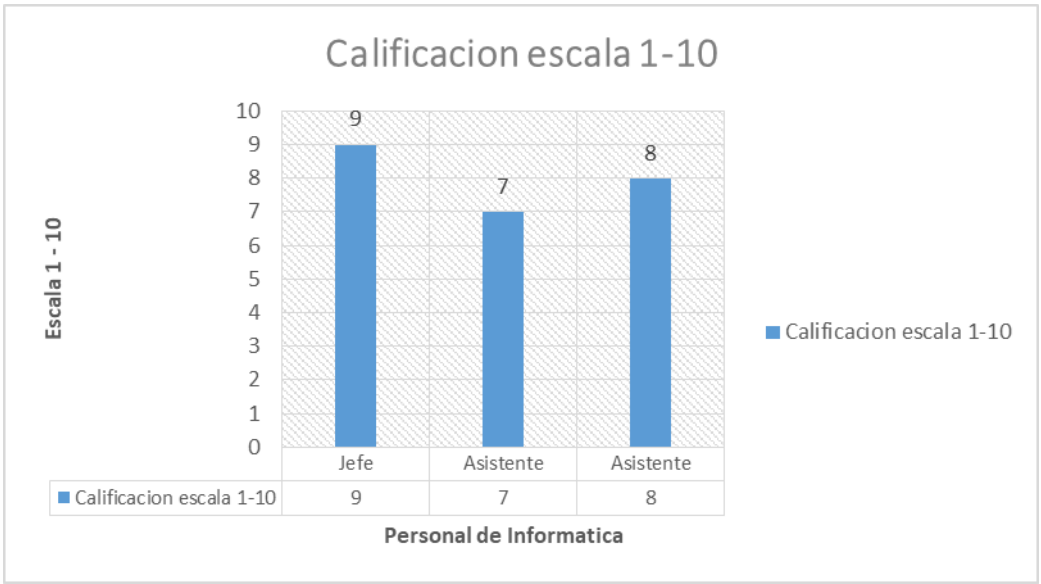

Ilustración 4: Calificación del personal de informática post implementación

Donde se nota una mejora sustancial del concepto del personal de informática hacia los servicios post implementación del proyecto.

Finalmente, observándolos resultados de la post implementación podemos afirmar que la implementación de una infraestructura de alta disponibilidad si disminuye las interrupciones en los servicios informáticos críticos mejorando la calidad del servicio hacia los usuarios garantizando los procesos de la Universidad que se respaldan en estas tecnologías.

Capítulo VI:

CONCLUSIONES

CAPÍTULO VI: CONCLUSIONES

1. Se analizó la situación actual de la infraestructura tecnológica referente al tipo de red LAN que usa, sistemas operativos que están implementados y planes de crecimiento descubriendo segmentos aislados y sin seguir estándares de instalaciones y categoría menores al mercado.
2. Se analizó los servicios informáticos que se brinda actualmente en la Universidad, en que están desarrollados y en que plataforma están funcionando actualmente Windows o Linux para tener una idea clara de su forma y modo de funcionamiento de sus servicios.
3. Se realizó un estudio comparativo de las mejores alternativas en el mercado de las soluciones en alta disponibilidad que se adecuen con los servicios informáticos brindados por la Universidad seleccionando la mejor alternativa.
4. Se logró configurar los servidores Windows Server 2012 y Linux Centos con las soluciones de Alta disponibilidad de Red creando cluster de Servidores, alta disponibilidad de disco y replica de Base de datos SQL Server y MySql, DNS y servicio de archivos usando software libre y propietario para luego integrarlos realizando pruebas de continuidad ante fallos.

Capítulo VII

RECOMENDACIONES

CAPÍTULO VII: RECOMENDACIONES

1. Capacitar al personal de la Oficina de Informática en cursos de especialización en administración de servidores de red para garantizar el buen funcionamiento las 24 horas de día de los equipos de comunicación y servidores.

2. Realizar un estudio de la red local y ver su estado para dar un mantenimiento preventivo o ya proyectarse a realizar un nuevo cableado estructurado de toda la Universidad, especialmente de la Oficina de Informática.

3. Tener un registro de fallas y métodos de que se emplearon para volver a poner en marcha los servicios, actualizando planes de contingencia y realizar pruebas de continuidad y contingencia bimestralmente para estar preparados ante futuras fallas.

4. Realizar un estudio del Cuadro de asignación de personal en la Oficina de Informática y proponer nuevos puestos de especialistas para contratar personal y así poder reforzar el equipo de trabajo la Oficina de Informática y lograr mejorar los servicios.

Capítulo VIII
PROPUESTA TECNOLÓGICA

CAPÍTULO VIII: PROPUESTA TECNOLÓGICA

8.1. Generalidades.

La Universidad se encuentra ubicada en el distrito de Pimentel, Provincia de Chiclayo Región Lambayeque. Actualmente cuenta con tres pabellones el cual se dedica exclusivamente a la educación superior universitaria dentro de la Región Lambayeque, y es por esto que se tiene que tener una integración rápida y segura de sus sistemas, con datos íntegros y seguros para poder dar un buen servicio a los estudiantes, contando con sistemas que estén disponibles 100% no solo en horarios de atención al público si no las 24 horas del día para atender peticiones y accesos a personal de la ciudad de Jaén y estudiantes a cualquier horario para accesos vía web a aplicaciones. Es por esto que el presente proyecto se centra en el mejoramiento de la continuidad de las aplicaciones y base de datos de los sistemas informáticos de la universidad. Este mejoramiento se hará a través de la implementación de tres servidores en alta disponibilidad en espejo para redundar el servicio de aplicación, base de datos y de archivos y DNS, permitiendo la continuidad de los servicios ante caídas en tiempo real, de manera eficiente y segura.

Actualmente, en la empresa no existe un plan de continuidad de los sistemas informáticos que respalde la caída de las aplicaciones o base de datos lo cual paralizaría los servicios de atención a los estudiantes y trabajadores en general lo cual generaría malestar en la población estudiantil y sus apoderados, motivo por el cual la implementación de servidores en alta disponibilidad de base de datos, aplicaciones, DNS y archivos es una propuesta para dar solución a este problema, obteniendo de esta manera seguridad y confiabilidad en los sistemas informáticos

para atender a los estudiantes mediante herramientas de software libre combinado con software propietario.

A continuación se muestra los servicios al implementar en alta disponibilidad:

- Servicio DNS
- Servicio WEB (Aplicaciones)
- Servicio Base de datos (Mysql y sql server)
- Servicio de Archivos

Estos servicios se implementaran en tres servidores

Servidor uno: Servicio DNS, dominio y Base de Datos Sql Server en discos espejos RAID 01 y duplicidad de tarjetas de red para dar redundancia a las conexiones de red.

Servidor dos: Servidor de aplicaciones (web con conexiona base de datos mysql) y servidor de archivos. También se implementara duplicidad de tarjetas de red para dar redundancia a las conexiones de red.

Servidor tres: Sera el espejo de los servicios del servidor dos.

8.2. Análisis

8.2.1. Análisis actual de los servicios de la empresa

La institución actualmente cuenta con los siguientes servicios locales como son un servidor de aplicación bajo la plataforma web que incluyen aplicación desarrollada en php que tiene acceso a base de datos en mysql, también se brinda el servicio de archivo bajo la plataforma Linux para accesos a los archivos de los usuarios y estos servicios solo tienen acceso solo en la sede principal. No cuenta con servidores de correo electrónico, usa uno de forma gratuita por ser institución educativa, tampoco se encontró en el análisis de la situación actual servidores de VoIp, usan telefonía convencional es más por cada facultad una línea distinta, es decir para llamadas entre facultades se realizan

llamadas locales, lo cual perjudica grandemente a la Universidad en sus costos. No existen planes de continuidad y ninguna infraestructura que respalde desastres naturales como de falla de los equipos.

8.2.2. Análisis actual de los servidores de la empresa

La institución cuenta con dos servidores en las cuales en uno de ellos corre el servicio web con sus aplicaciones locales conectadas a diversas bases de datos en el sistema operativo Linux, además de ser servidor de archivos para algunas áreas.

El segundo servidor es usado para dar acceso remoto a trabajadores ubicados en la sede de Jaén que no es parte de este estudio, corre bajo el sistema operativo propietario Windows.

El tercer Servidor es de DNS que funciona bajo el sistema operativo Windows además de la base de datos en SQLserver y algunas aplicaciones antiguas en lenguaje de programación Clipper.

No existe actualmente algún servidor que haga de controlador de dominio o haga de servidor para reutilizar equipos con bajos recursos de hardware. Se pudo encontrar el dato del 100% de equipos de cómputo el 30% de ellos son de bajos recursos de hardware que a pedido de los usuarios dejarían de funcionar lo cual acarrearía un gasto innecesario a la Universidad de Chiclayo y afectaría su presupuesto.

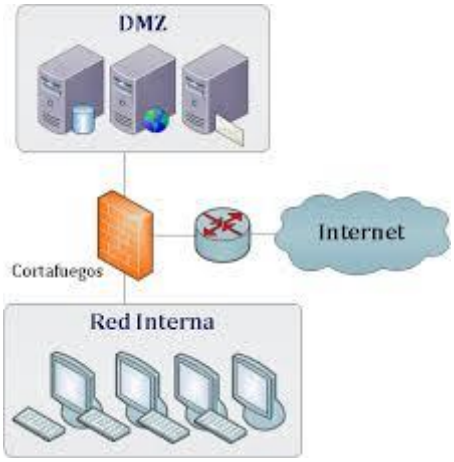

Ilustración 5: Diagrama lógico de la red actual

Fuente: Propia

8.2.3. UBICACIÓN DE LAS OFICINAS

Actualmente la Universidad cuenta con un solo local ubicado en la Carretera a la Ciudad de Pimentel, dispone de tres edificios: oficinas centrales, y dos pabellones de aulas con cada una de 4 pisos. Dentro de ambos pisos existen varias facultades. En este proyecto tendremos en cuenta solo la oficina de Informática específicamente donde están becados la sala de servidores.

Ilustración 6: Oficina Central

Ilustración 7: Dos pabellones de aulas

Ilustración 8: Vista satelital de Universidad de Chiclayo

La red de comunicación se dejará preparada para una futura ampliación que contemple las demás delegaciones en un segundo proyecto.

Para localizar más fácilmente las oficinas las llamaremos a las Pabellón 01, Pabellón 02 y Oficina Central.

8.3. Diseño

A continuacion se muestra el diseño logico propuesto para dar solucion a la hipotesis sugerida, con la conexión e implementacion de los tres servidores y la conexión redundante de tarjetas de red de cada uno.

Se plante a realizar un diseño de una infraestructura de red en alta disponibilidad que garantice la continuidad del servicio implementando tres servidores uno con discos espejos y otros dos uno activo y otro pasivo el cual entrara en funcionamiento en caso de caída del servidor activo, estos servidores deberán ser instalados y configurados haciendo uso del sistema operativos Linux y Windows.

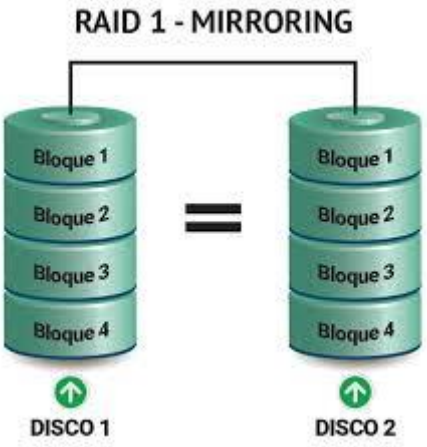

Ilustración 9: Diseño de Servidor Windows con discos espejos en Raid 1

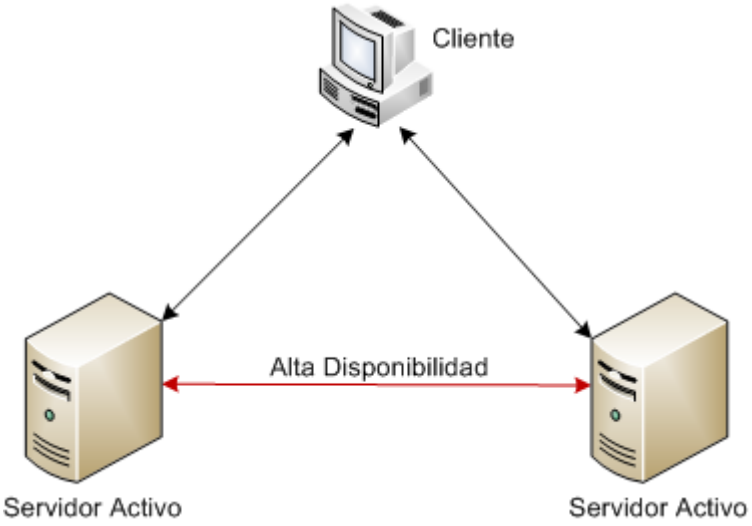

Ilustración 10: Diseño lógico propuesto de servidores Linux en clúster

Fuente: https://paucls.wordpress.com/2010/02/13/clusters-de-alta-disponibilidad-ha/

El proyecto sugiere (no es parte del proyecto) la total renovación de los switchs de la Sala de servidores de la Oficina de Informática de la Universidad, dando conexión a servidores usando conexiones redundantes, usuarios finales (PCs y Teléfonos IP) y disponiendo de puertos de enlace para la interconexión a las demás oficinas de la misma ciudad universitaria.

• Sistema de Telefonía IP: aunque el presente estudio no abarca este tema o implementación de esta tecnología servirá de base para esta solución tecnológica que bajaría los costos en forma abrumadora en la parte de la telefonía.

• Cableado: Contempla el cableado necesario para interconectar los switchs dentro de la Sala de Servidores de la Oficina de Informática.

• Alimentación hardware de red: Contempla la instalación de SAI's para la alimentación de la electrónica de red LAN. Los equipos de voz is van aprovisionados con baterías propias del fabricante.

En nuestro caso usaremos switchs gestionables que permiten su configuración. Todo esto nos aportara mayor control, rendimiento y eficiencia sobre la red de cliente.

Switch de acceso

Los switch de acceso se usarán para permitir el acceso a la red a los usuarios finales. Pertenece a la capa donde los usuarios finales se conectan a la red. Los switches de la capa de acceso por lo general ofrecen conectividad de Capa 2 (VLAN) entre los usuarios. Los dispositivos de esta capa deben tener estas opciones:

• Coste por puerto de switch bajo

• Alta densidad de puertos

• Escalabilidad de uplinks a capas superiores

• Funciones de acceso al usuario como VLANs, filtrado de tráfico y protocolos y Calidad de servicio (QoS)

• Capacidad a través de múltiples uplinks

• Local VLANs

Switch de núcleo

Esta capa ofrece conectividad a los dispositivos de la capa de distribución. En nuestro caso no tenemos switch que se dediquen únicamente la distribución porque nuestra red es pequeña y queremos aprovechar todos los switch para conectar usuarios.

El core, también llamado backbone, debe ser capaz de mover tráfico los más eficientemente posible. Los dispositivos de esta capa deben tener estas opciones:

• Muy alto rendimiento en la Capa 3.

• Manipulación de paquetes sin coste ni penalización alguna (listas de acceso, filtrado de paquetes)

• Redundancia y capacidad para alta disponibilidad.

• Funciones avanzadas de calidad de servicio (QoS).

Descripción del hardware contemplado

Nuestra red LAN estará basada en la interconexión de los diferentes equipos switch de acceso y de núcleo o CORE de la red mediante fibras.

Finalmente nuestra red Lan debería quedar algo como la siguiente figura

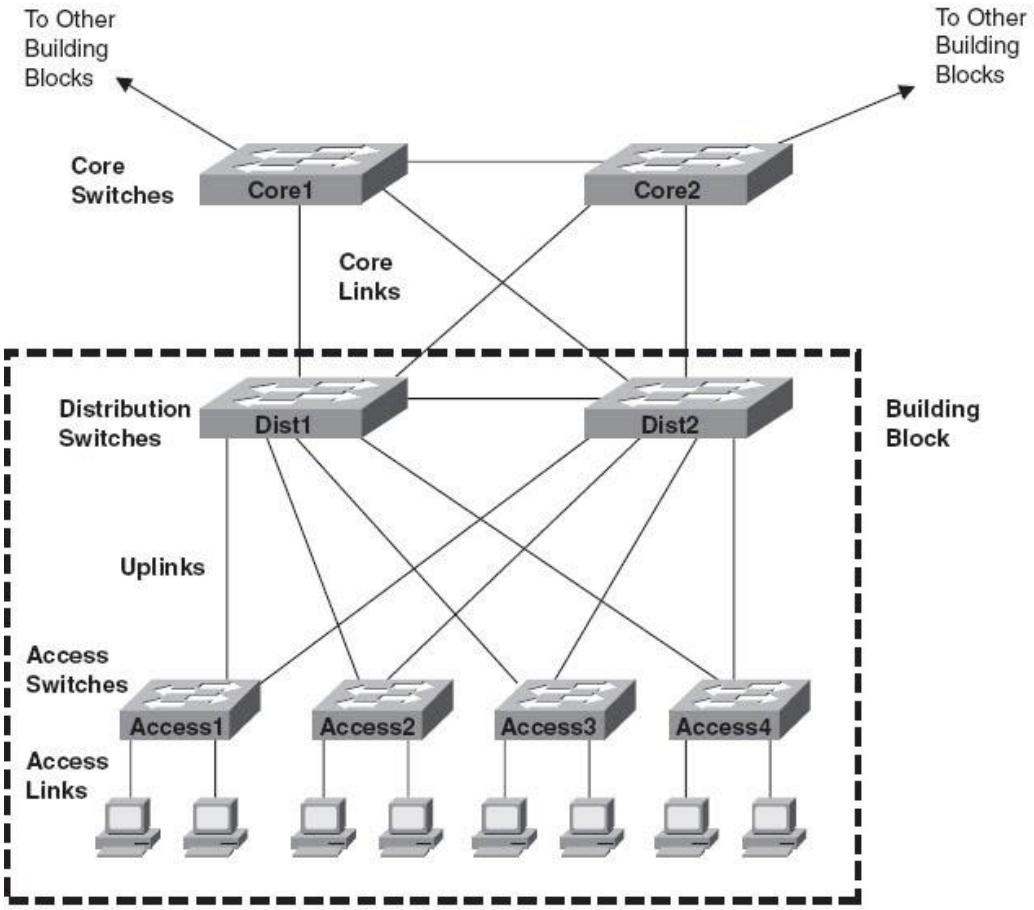

Ilustración 11: Diseño lógico propuesto de equipos de comunicación

Fuente: https://paucls.wordpress.com/2010/02/13/clusters-de-alta-disponibilidad-ha/

Es decir switch core redundantes ubicados en la Oficina Central y swicth de Distribución y de acceso con conexiones redundantes en la oficina central y pabellones 01 y 02.

Recordar que los switch de acceso son directo a los puntos de red.

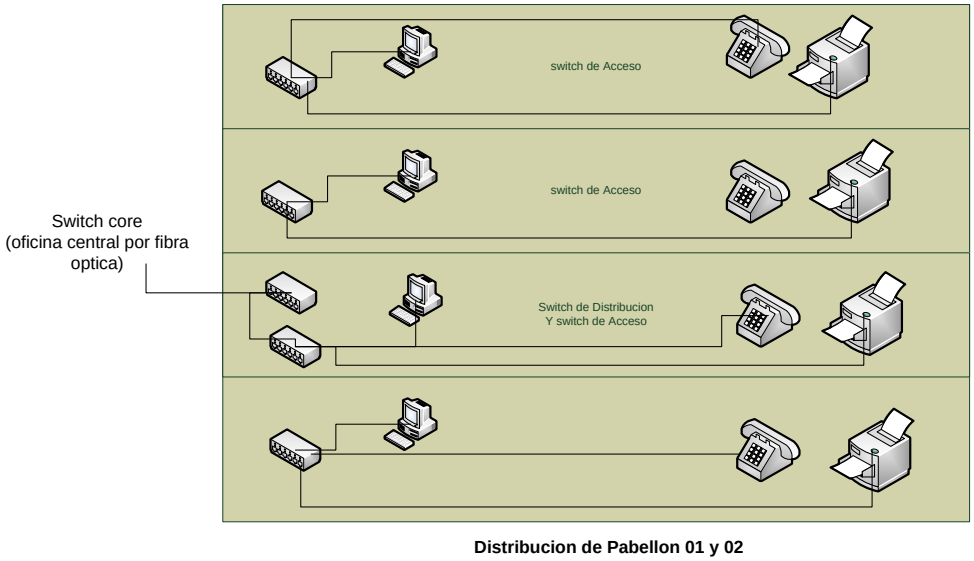

Ilustración 12: Diseño lógico propuesto de switch de distribución

Fuente: https://paucls.wordpress.com/2010/02/13/clusters-de-alta-disponibilidad-ha/

La figura anterior muestra la distribución de la red en pabellón 01, 02 y central en los cuatro pisos ubicando los switch de distribución y de acceso.

8.4. Implementación

Para la implementación de nuestra solución utilizaremos el Sistema Operativo Centos 6 en el que instalaremos las base de datos Mysql server con servidor de aplicaciones Apache con soporte a PHP y el servicio samba como servidor de archivos, estos servidores deberán estar instalados y configurados dentro de un cluster de alta disponibilidad. También se instalara el sistema Operativo propietario Windows server 2012 en que se instalara sql server DNS y el servicio de Dominio. En general utilizamos tres nodos del servidor de la base de datos y aplicaciones.

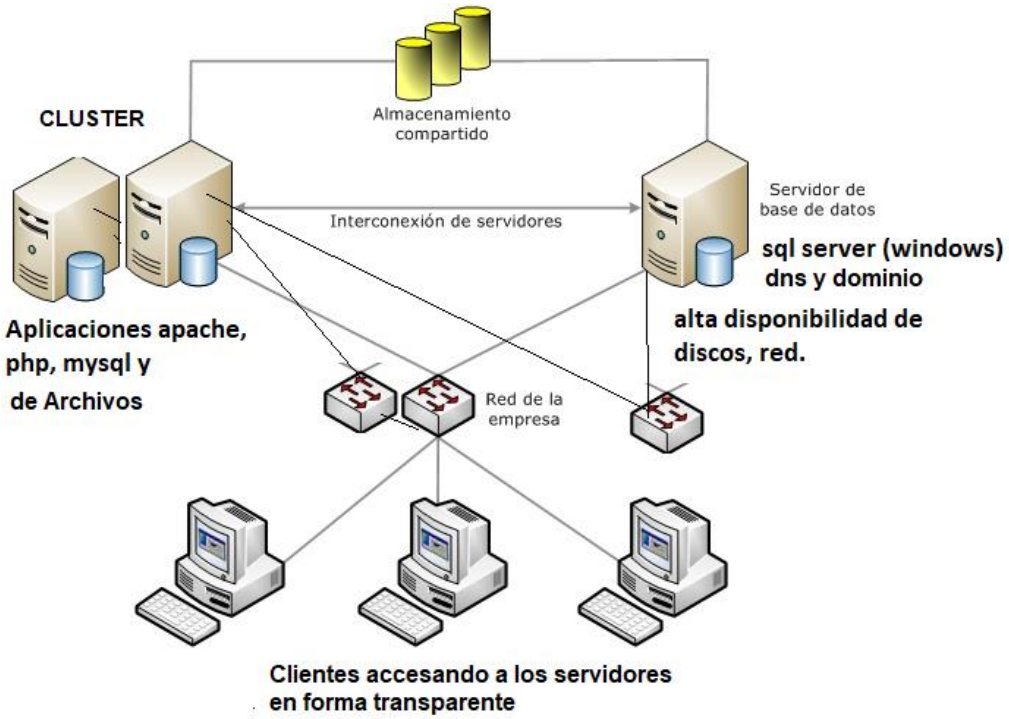

Ilustración 13: Propuesta de Implementación de servidores en alta disponibilidad

8.4.1. Instalación del equipamiento lógico necesario.

Distribución lógica de IP para servidores y aplicaciones a instalar

Servidor uno:

Nombre	: nodowin
IP	: 192.168.0.5 / 255.255.255.0
Puerta enlace	: 192.168.0.1
Servicios	: DNS / Dominio / Sql Server
Tipo alta disp.	: Discos y tarjetas de red
Numero de discos	: 02
Tarjetas de red	: 02

Servidor dos:

Nombre	: nodoapp1
IP	: 192.168.0.6 / 255.255.255.0
Puerta enlace	: 192.168.0.1
Servicios	: Aplicaciones (web + mysql) y archivos

Tipo alta disp. : Servidor Activo (cluster)
Numero de discos : 01
Tarjetas de red : 02

Servidor tres:

Nombre :nodoapp2
IP : 192.168.0.7 / 255.255.255.0
Puerta enlace : 192.168.0.1
Servicios : Aplicaciones (web + mysql) y archivos
Tipo alta disp. : Servidor Pasivo(cluster)
Numero de discos : 01
Tarjetas de red : 02

IP virtual para la administración del cluster de alta disponibilidad

Nombre : virtual1

IP : 192.168.0.111 / 255.255.255.0

Aplicaciones :

Administración de servidor de Balanceo de Carga de nodos web y administrador del cluster de base de datos mysql.

Este servidor virtual instalado en la pc Windows server.

La disponibilidad de alta disponibilidad como para recordar permite la conexión de un grupo de servidores (llamados nodos o miembros) para funcionar juntos como un clúster.

8.4.2. Instalación en CentOS.

- La instalación de Sistema operativo Centos se hará en los nodos nodoapp1 y nodoapp2 que servirá de servidor de aplicaciones. Los pasos serán simples como insertar el DVD, asegurar que el equipo arranque a través del DVD y proceder a la instalación dando ENTER al

mostrar la pantalla inicial (Opción: Install or upgrade an existing System)

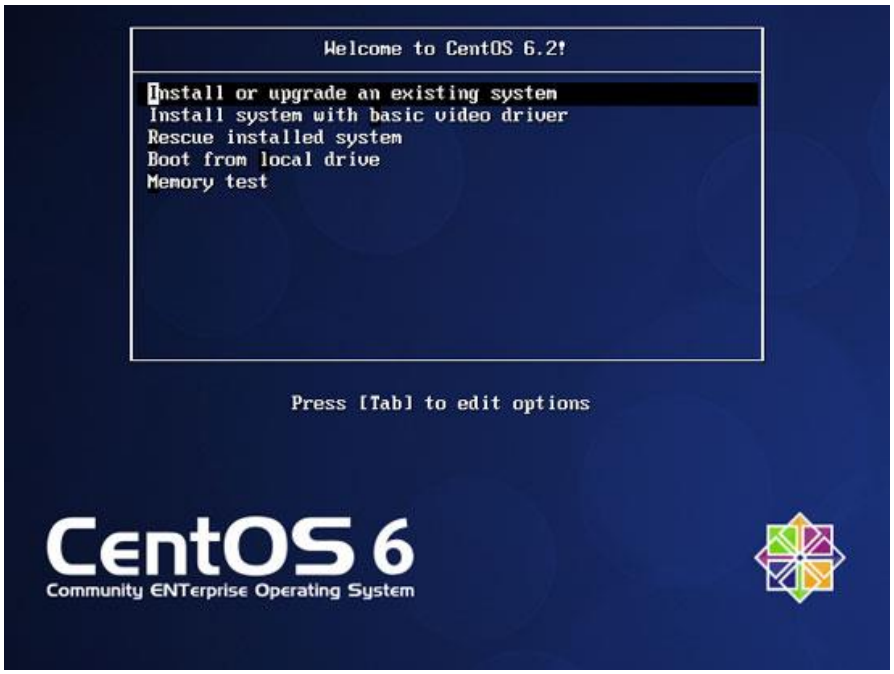

Ilustración 14- Pantalla de inicio de instalación

Ilustración 15- Inicio de instalación

- Finalmente la instalación se completa y se debe reiniciar el servidor sin el DVD:

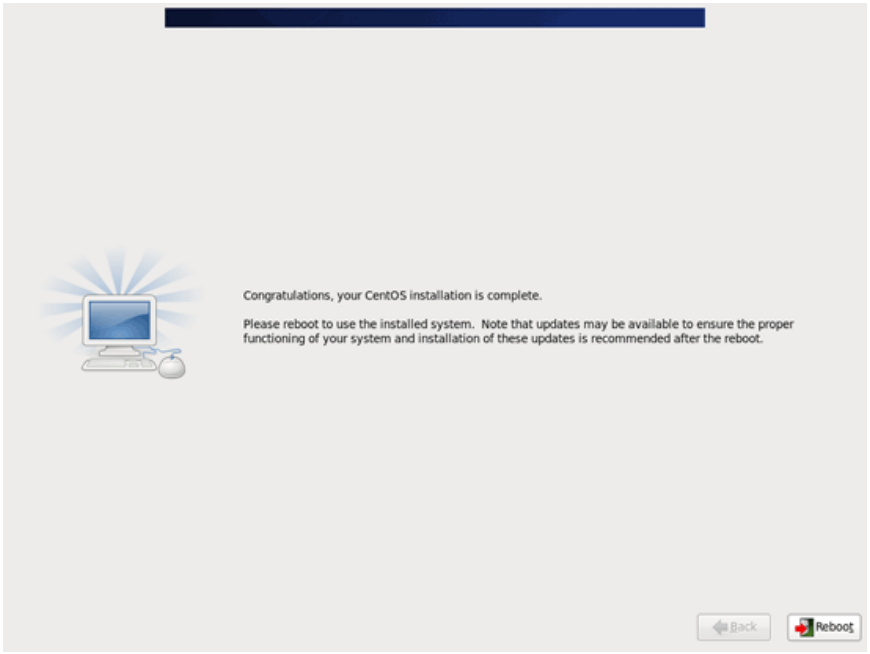

Ilustración 16- Reiniciar servidor

- Configuración de las interfaces de red.

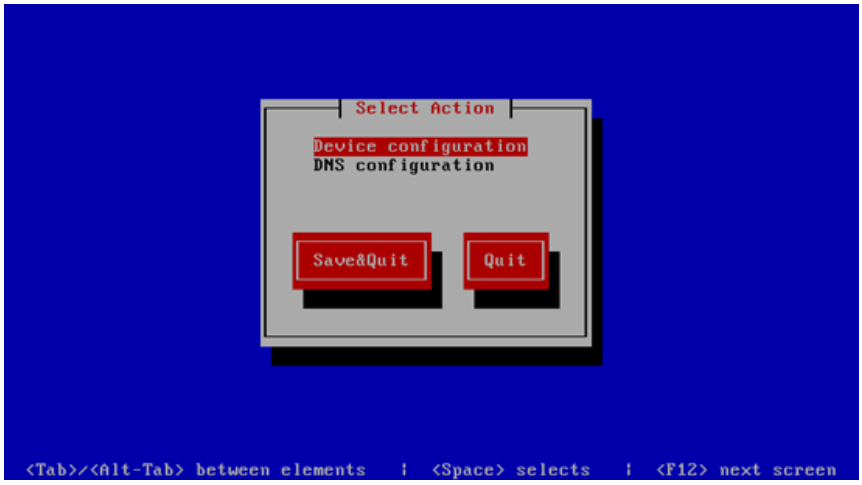

Ilustración 17- seleccionar dispositivo

- Se selecciona la interfaz de red:

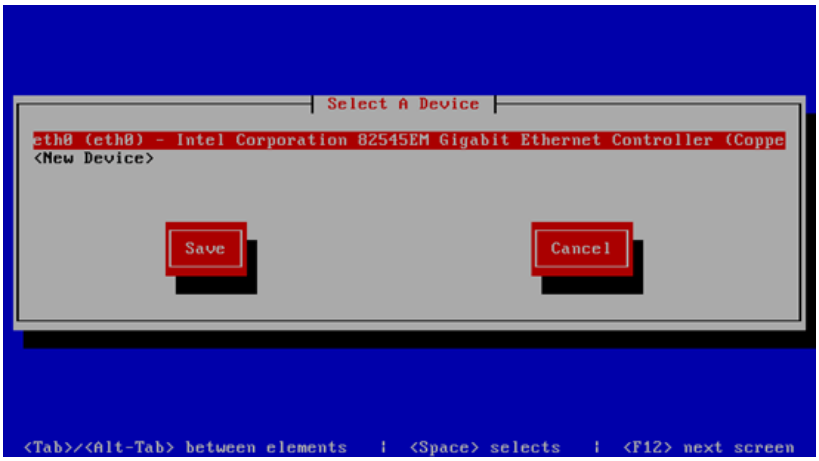

Ilustración 18- Seleccionar tarjeta red

- Llenar los detalles de la red desmarcando la opción DHCP y colocando manualmente la IP Estática, Máscara de Red, Puerta de Enlace y los DNS de acuerdo el cuadro anterior, solo para los nodos nodoapp1 y nodoapp2.

- Luego seleccionar “Save”

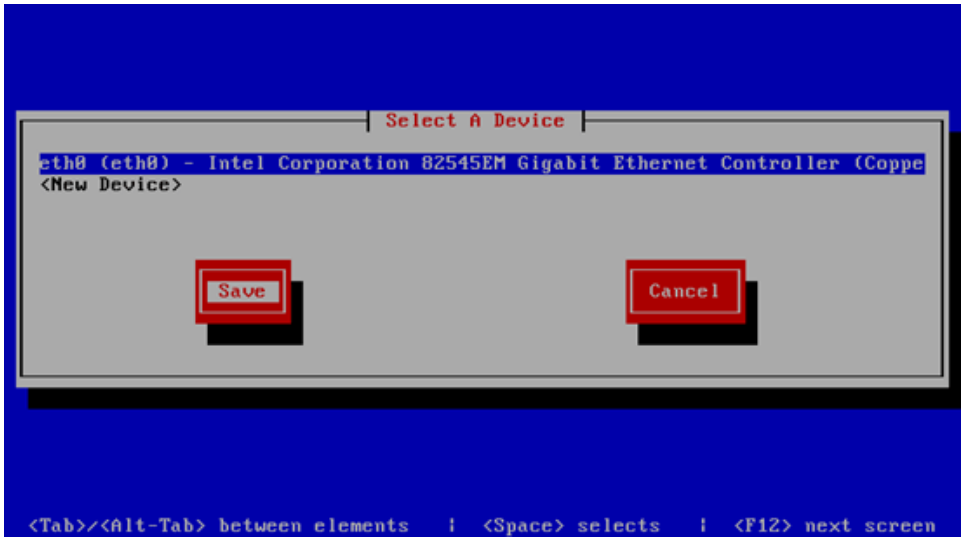

Ilustración 19- agregar cambios de eth0

- Ahora se pueden agregar DNS adicionales, seleccione “DNS Configuration”

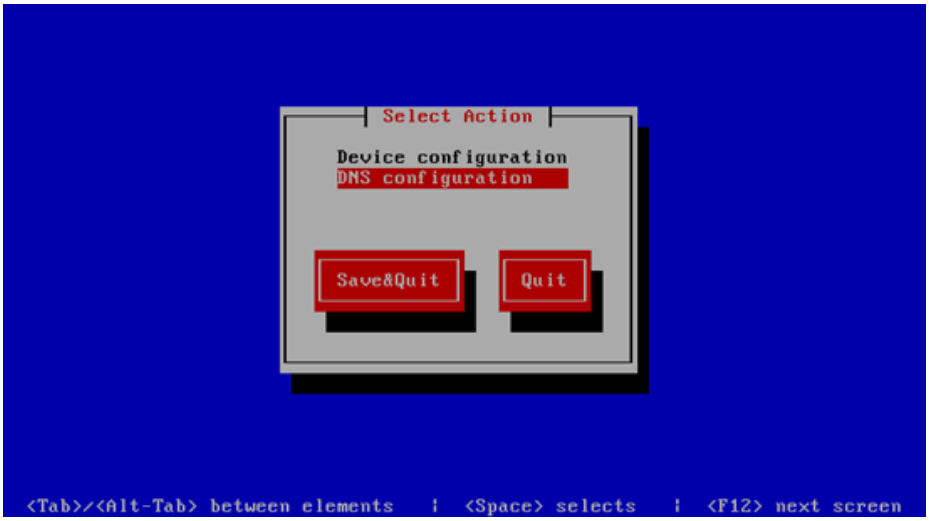

Ilustración 20- Configurar DNS

- Ahora podemos agregar los DNS 192.168.0.5. Que es el servidor local DNS instalado en Windows Server.

- Luego hacer clic en “Save&Quit”

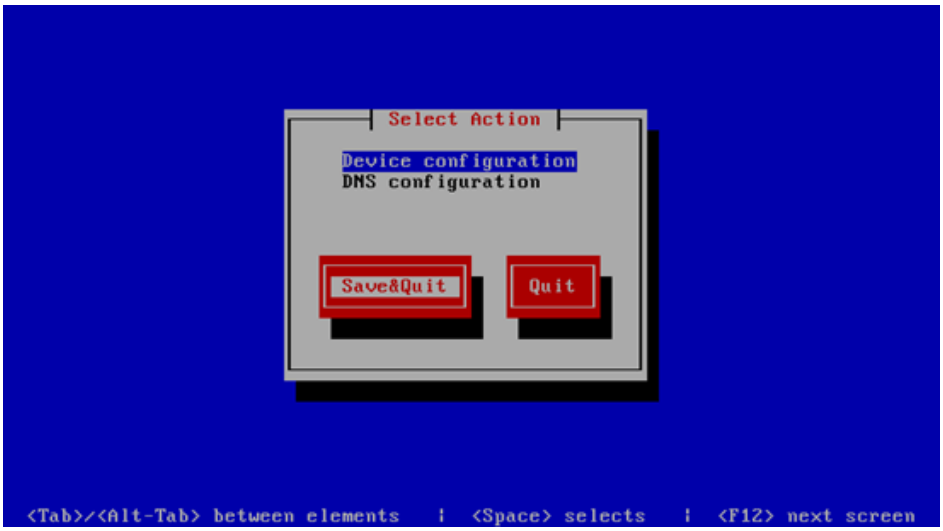

Ilustración 21- Grabar cambios hechos en interface de red

- Ahora debemos ejecutar el comando **ifconfig** verificar que el instalador obtuvo la IP correctamente, recuerden que deben de realizar este paso en los nodos nodoapp1 y nodoapp2.

```
[root@nodoapp1 ~]# ifconfig
```

```
eth0      Link encap:Ethernet  HWaddr 00:18:F3:18:9D:EF
inet addr:192.168.0.6 Bcast:192.168.0.255  Mask:255.255.255.0
```

- Debemos verificar el archivo /etc/resolv.conf par aver si tiene los DNS que colocamos:

```
[root@nodoapp1 ~]# cat /etc/resolv.conf
Nameserver 192.168.0.5
nameserver 8.8.4.4
Nameserver 8.8.8.8
```

Si por cualquier razón no los tiene podemos editar dicho archivo y agregarlos ó ejecutar:

```
# setup
```

8.4.3. Instalación en Windows Server 2012.

Para instalar en Sistema Operativos Windows server 2012 insertamos el DVD de instalación del sistema operativo y nos mostrara una pantalla como en la figura siguiente:

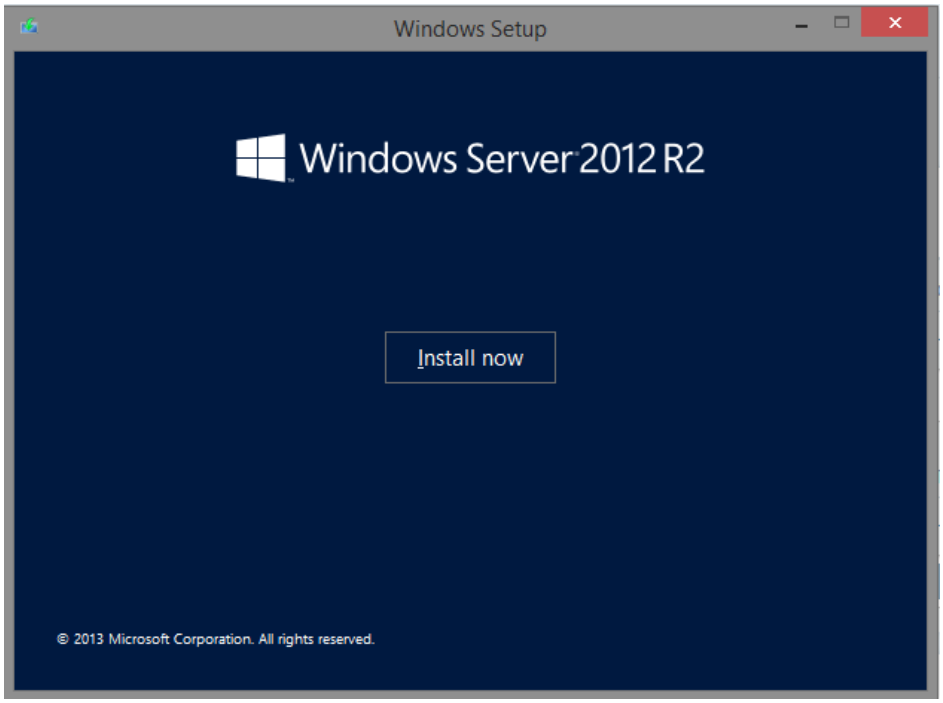

Ilustración 22- Pantalla inicial de instalación de Windows server 2012

Seleccionamos el tipo instalación, sea modo gráfica y con todos los roles.

Otro paso importante seleccionamos la partición donde deseamos instalar el sistema operativo.

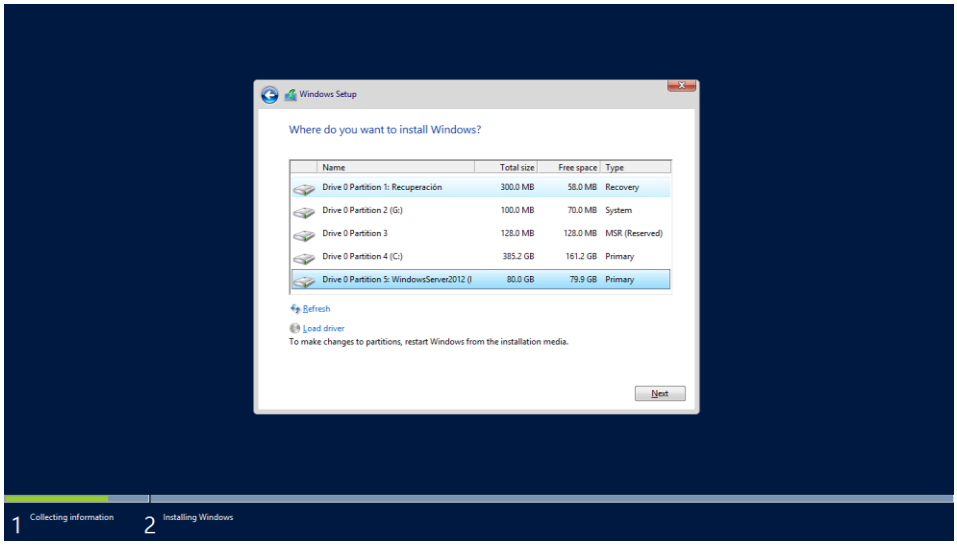

Ilustración 23- Selecciona la partición de disco donde se instalara

Luego iniciara el proceso de instalación copiando los archivos necesarios, esto puede tardar algunos minutos, esperaremos que termine de instalar.

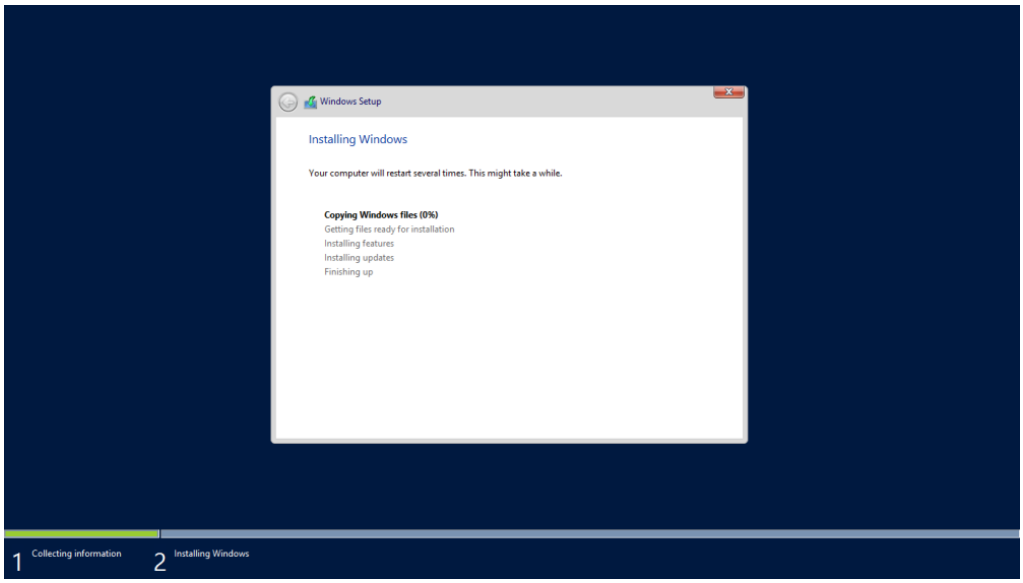

Ilustración 24- Proceso de instalación

8.4.4. Configuración de balanceo de interfaces de red en Windows Server.

Hay que recordar que el servidor Windows cuenta con dos tarjetas de red y que realizaremos balanceo de carga para dar mayor seguridad ante caídas en la red. El servidor Windows cuenta con dos tarjetas de red de 10/100/1000 que serán configuradas para ser balanceados.

Para realizar el balanceo de carga de la interfaces de red del servidor Windows usaremos la herramienta NIC Teaming que viene integrada con el sistema operativo e ingresamos a ella desde la consola de administración del servidor.

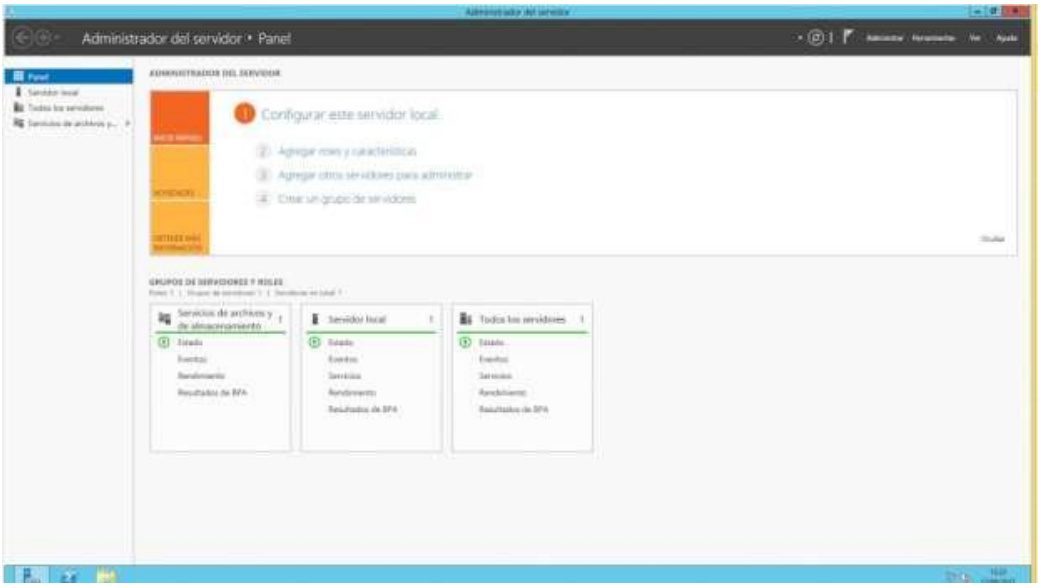

Ilustración 25- Consola de administración del servidor

Y seleccionamos la opción de servidor local y en la parte derecha se mostrara dentro de las propiedades la opción NIC Teaming que por defecto viene deshabilitada y nos aparecerá la lista de adaptadores de red y seleccionamos ambos.

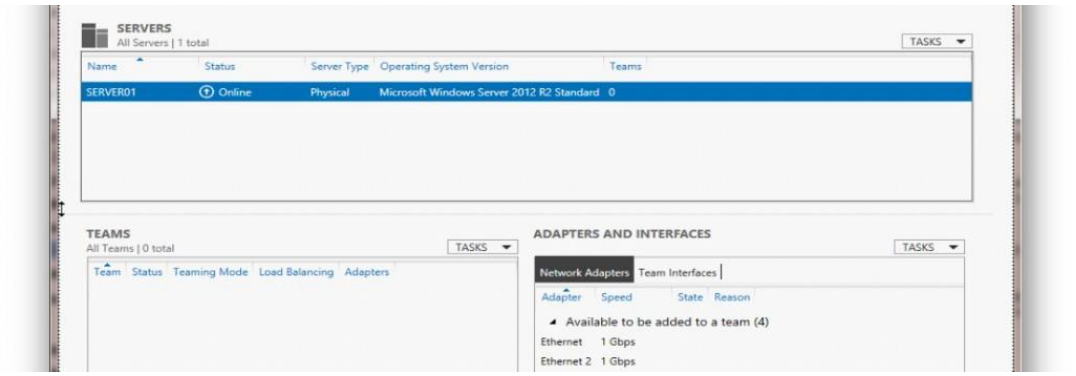

Ilustración 26- Propiedades de NIC Teaming

Seleccionamos ambos adaptadores (con Click y CTRL+Click) y en la opción task creamos una nuevo team, le damos el nombre y está listo.

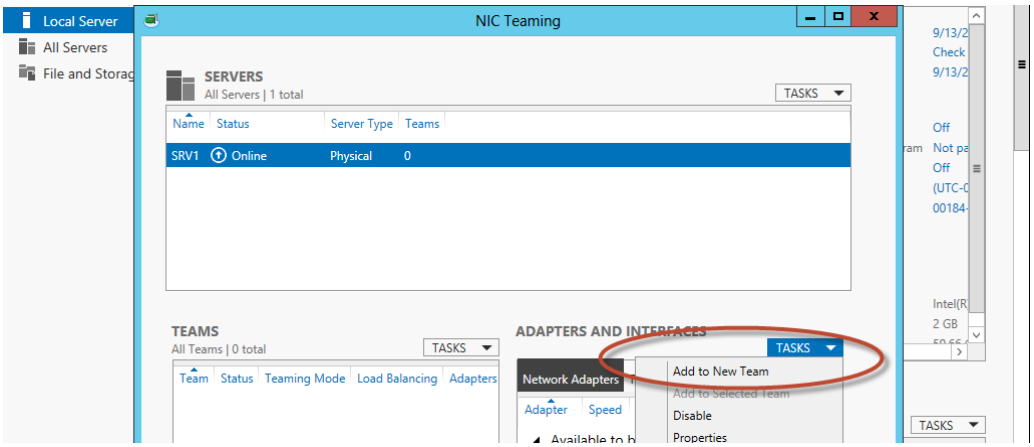

Ilustración 27- Adición de nueva tarea de NIC Teaming.

Nos debe quedar la pantalla así

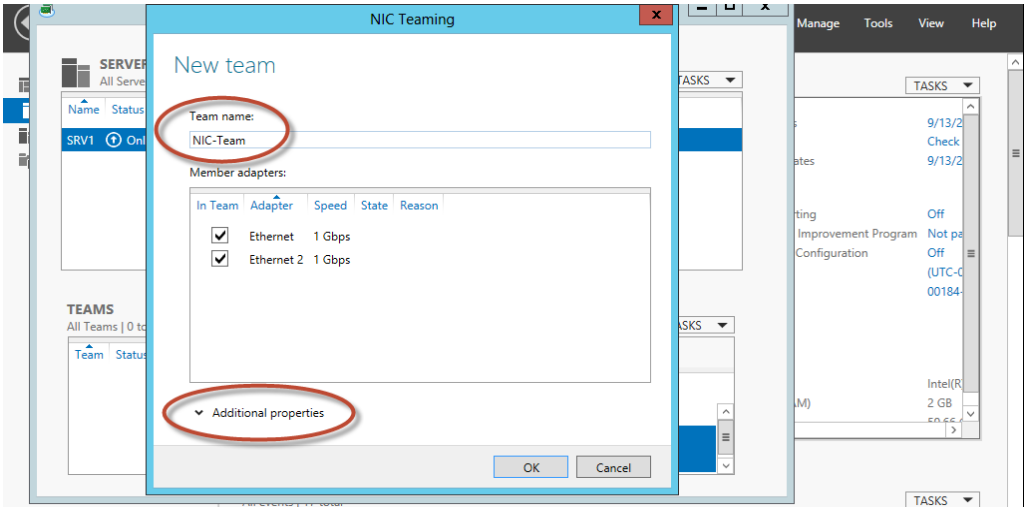

Ilustración 28- Selección de 02 tarjetas de red para formar el nuevo team

Ahora vamos a nuestra conexione de red, y nos aparecerá nueva interface de red.

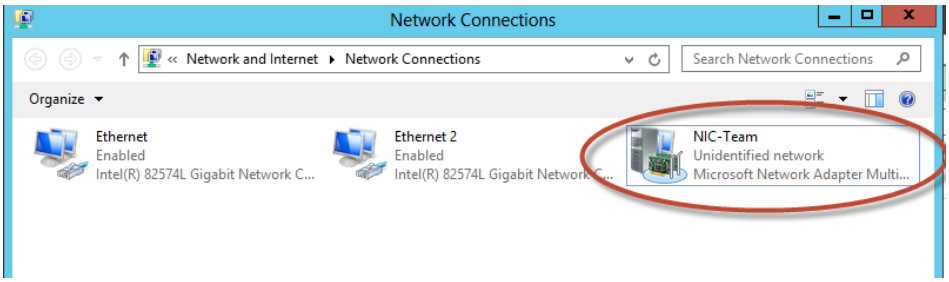

Ilustración 29- Nueva interface de red NIC Team

Solo nos quedaría asignarle los parámetros de red y listo para su configuración.

8.4.5. Montar disco espejo en servidor Windows nodowin

Existen varias formas de implementar discos espejos y una de ellas es raid 1, y en esta configuración lo haremos desde software haciendo uso de la herramienta instalada en el mismo sistema operativo. Otra opción es hacerlo desde hardware o BIOS con los controladores del disco, pero para efectos del este proyecto solo usaremos la primera opción teniendo en cuenta que se tiene instalado dos disco de iguales características que es lo primordial para este paso.

Para montar un RAID 1 en Windows solo tenemos que abrir el administrador de discos.

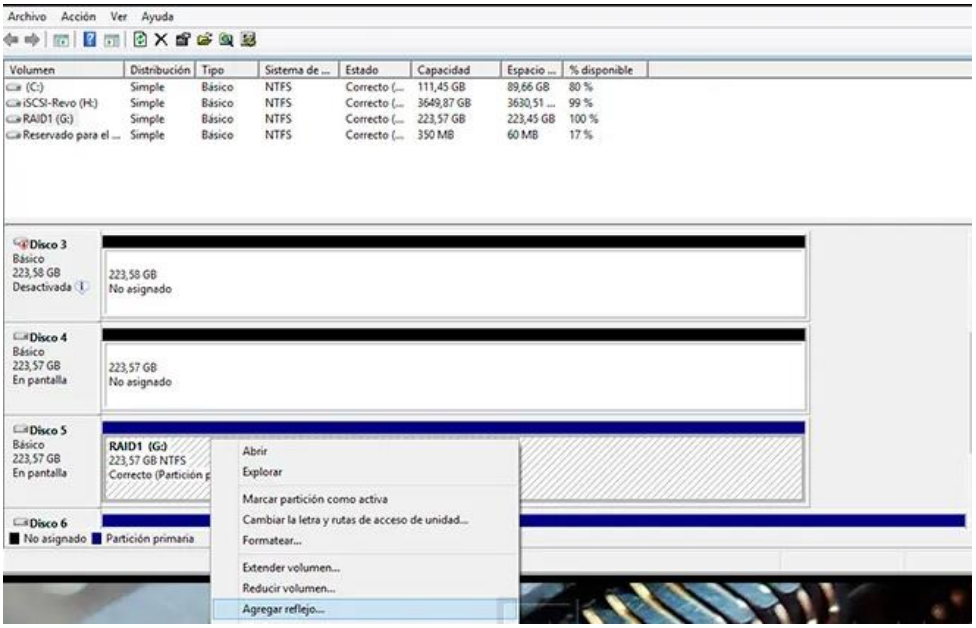

Ilustración 30- Creación de Raid 1

Como se muestra en la figura anterior solo debemos abrir el "administrador de discos" seleccionamos la partición que será disco espejo, damos click en el botón derecho y usar la opción "agregar

reflejo" y nos quedara de la siguiente forma Raid1 como un solo disco físico.

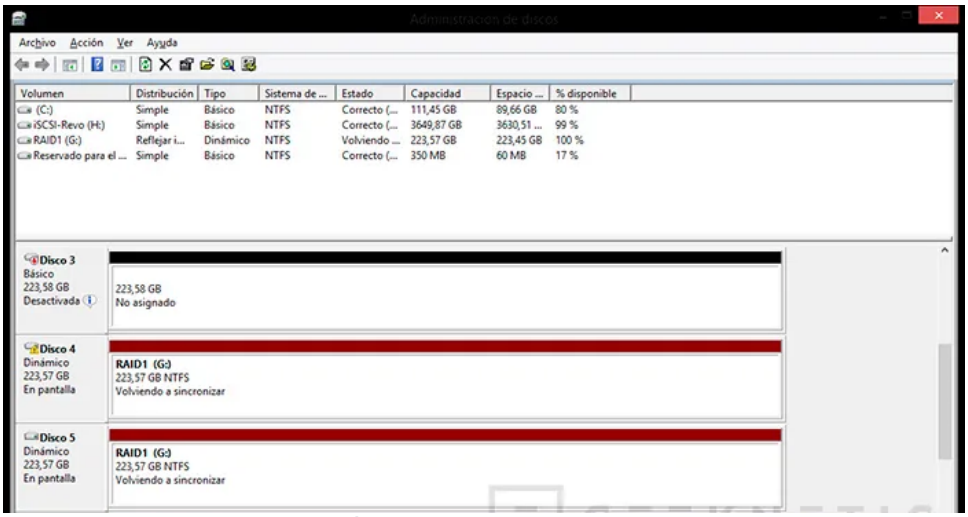

Ilustración 31- Disco en Raid 1

8.4.6. Implementación del servicio DNS en Windows server

Para la implementación del servicio DNS entramos a la ventana de configuración de servidor y seleccionamos agregar roles, y dentro de la lista de roles debemos seleccionar el rol o servicio DNS.

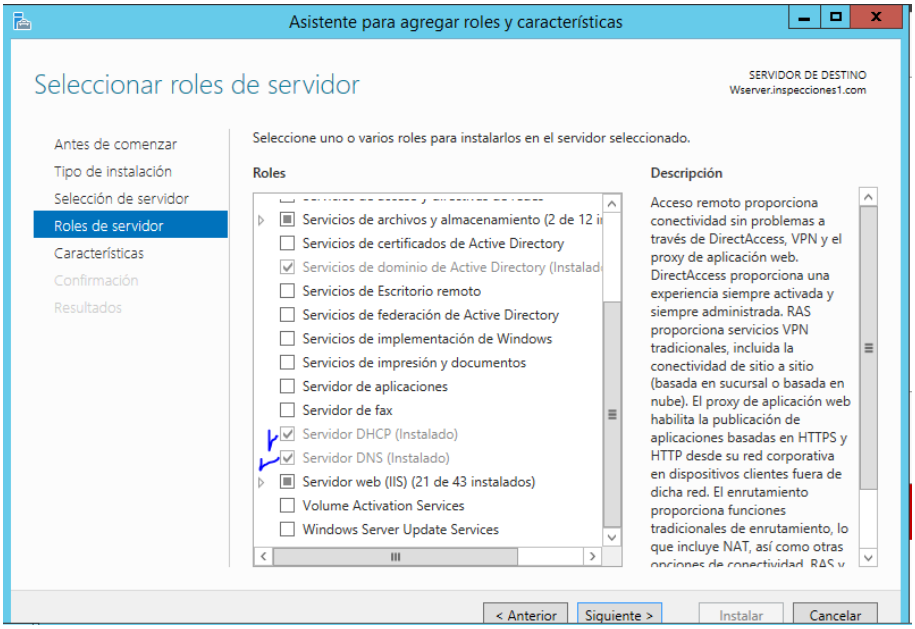

Ilustración 32 Seleccionamos rol DNS

Luego al finalizar la instalación se mostrará la ventana con las zonas de búsqueda directa e indirecta. Verificamos las zonas de búsqueda directa que se implementaron en el presente proyecto.

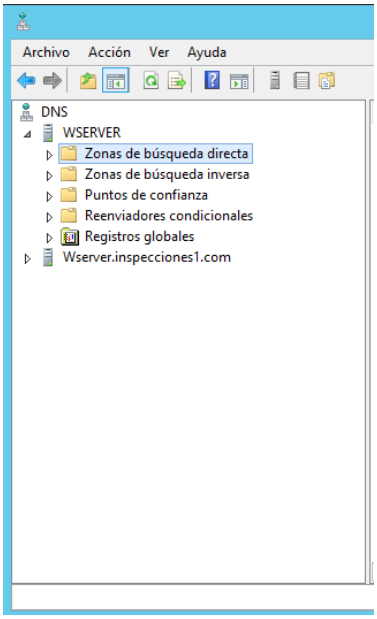

Ilustración 33Zona de búsquedas

Ahora procedemos a crear una nueva zona, lo realizaremos con el asistente dando click derecho en nombre del servidor /zona búsqueda directa /zona nueva

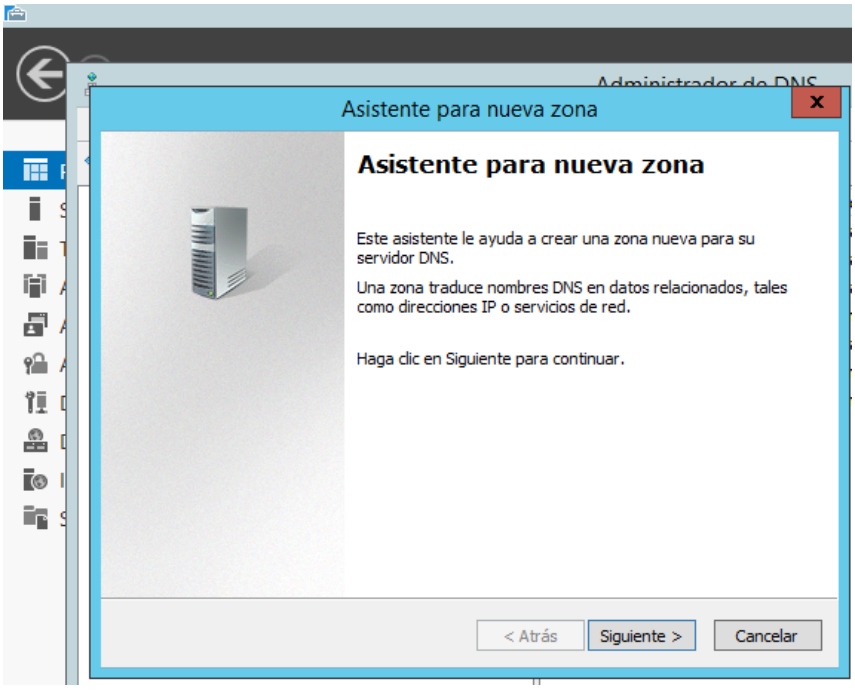

Ilustración 34 Asistente para crear nueva zona

Seleccionamos el tipo de zona a crear, en nuestro caso zona principal

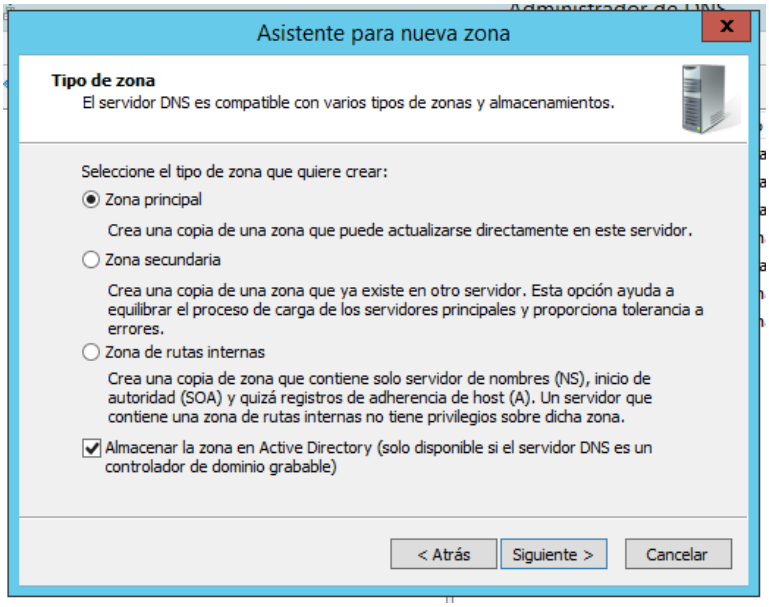

Ilustración 35Asistente creación de zonas

Escribimos el nombre de la zona a crear que coincida con el dominio creado, en nuestro caso colocaremos **udch.edu.pe**

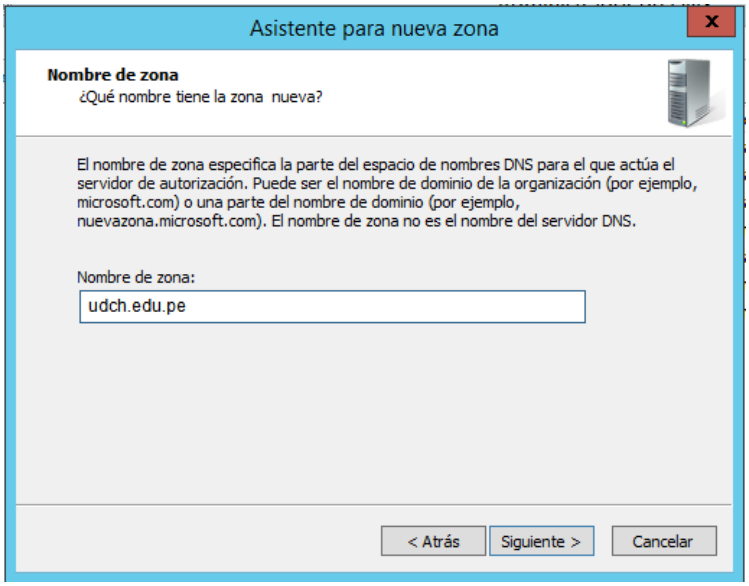

Ilustración 36 Ingresamos el nombre de zona

Permitir actualizaciones de nuestra zona de busqueda directa.

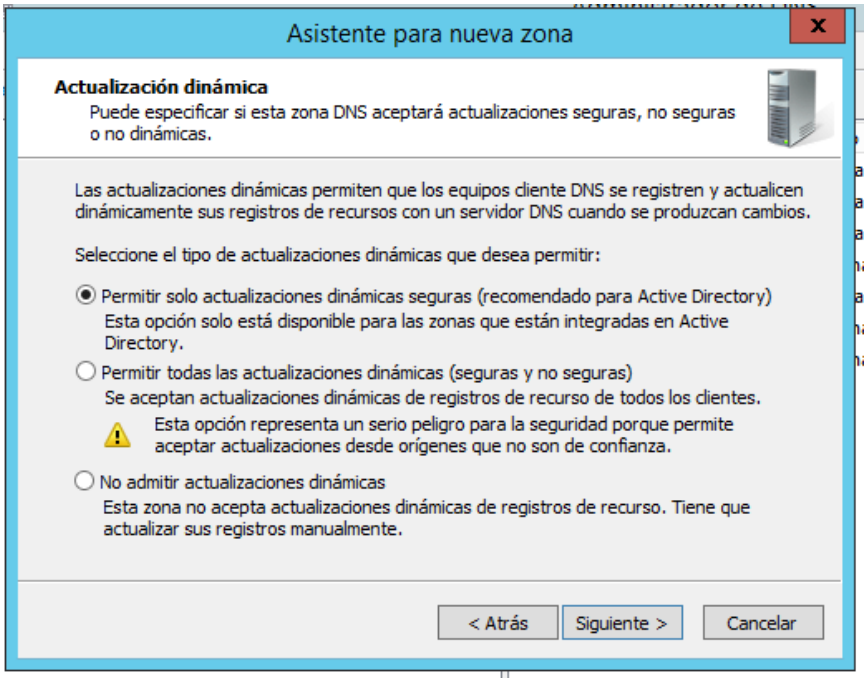

Ilustración 37 Permitir actualizaciones

Finalizamos la instalación de la zona de búsqueda directa

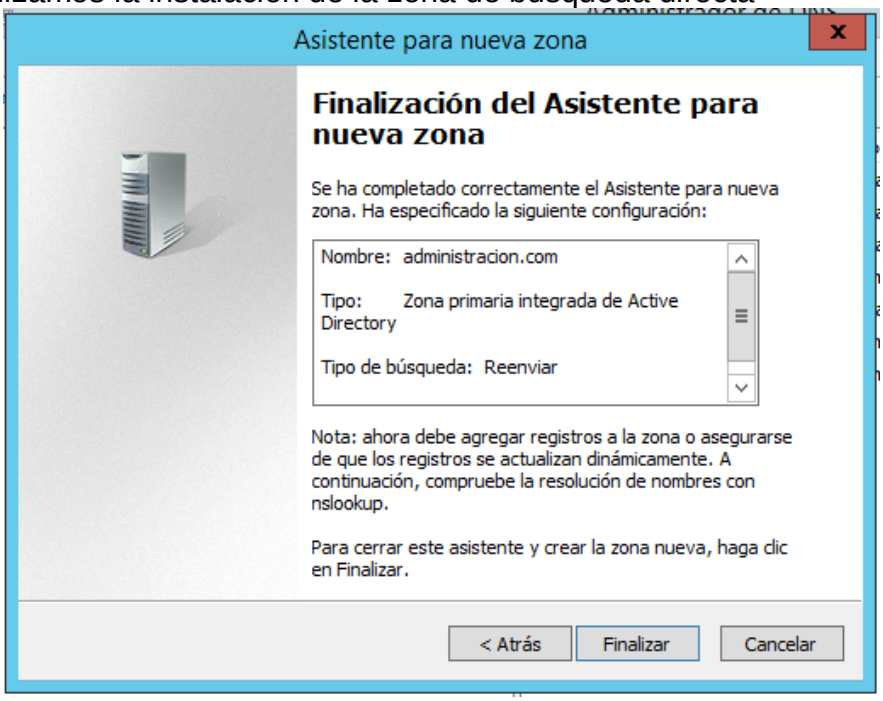

Ilustración 38 Fin de instalación de zona

Podemos observar en la lista de zonas de búsqueda directa nuestra zona creada udch.edu.pe.

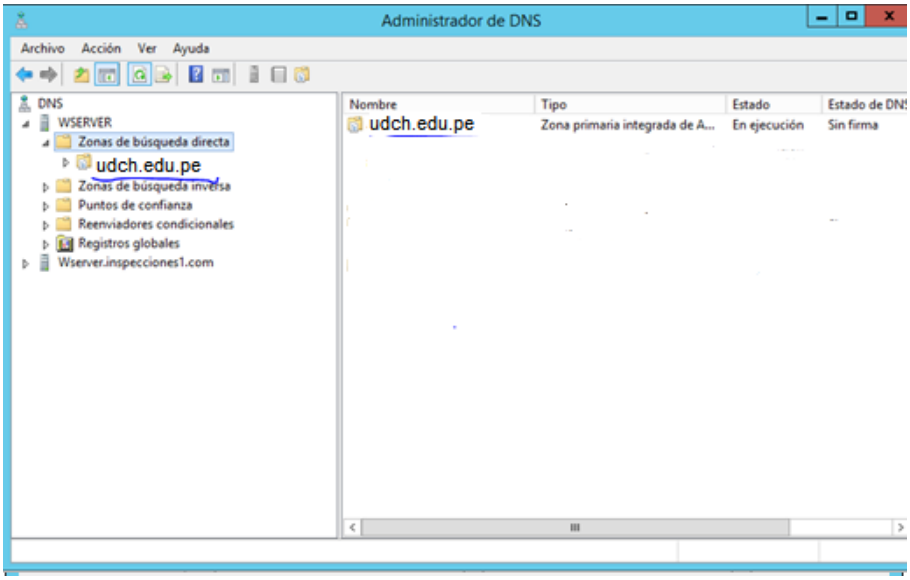

Ilustración 39Zonas directas

Luego iniciamos el nuevo procedimiento para configurar la zona de búsqueda con el asistente, dando click derecho sobre la nueva zona creada udch.edu.pe y seleccionamos host nuevo y colocamos nombre de los server y sus is para los servidores centos y Windows server.

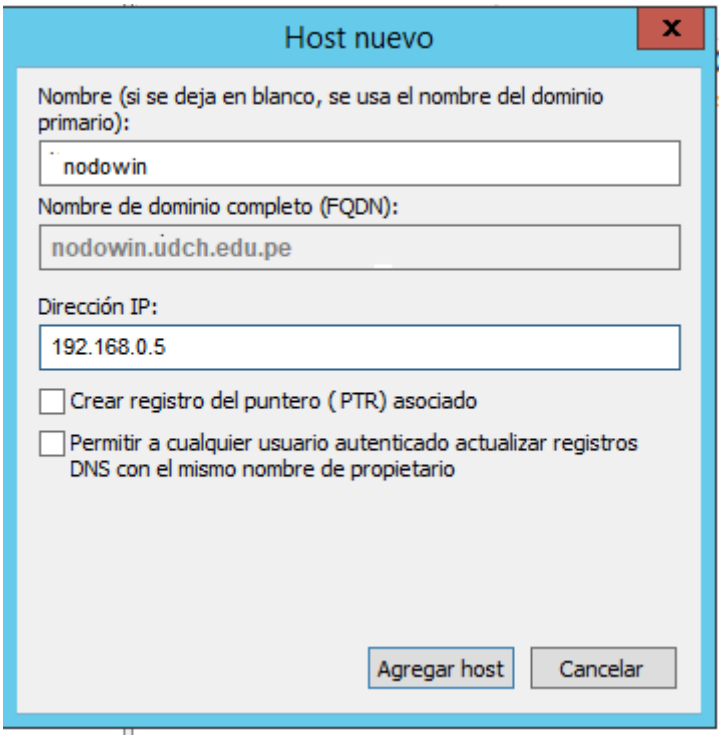

Ilustración 40 nombre de hosts

Los mismos pasos debemos hacer para los nodos de Linux centos es decir nodoapp1 y nodoapp2 con sus respectivos ip.

Comenzamos a configurar la zona de búsqueda, dando click derecho sobre la nueva zona y seleccionamos alias nuevo y colocamos nombre de los server y su alias www

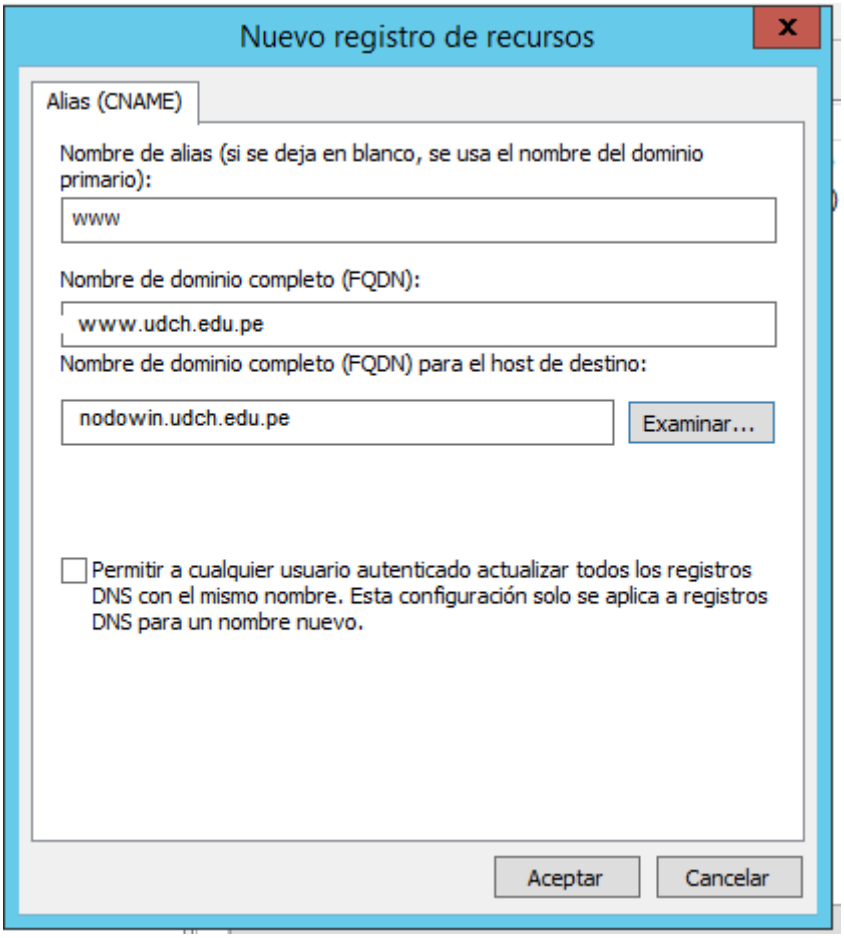

Ilustración 41 Alias que sera reconocido equipo

Hay que recordar que en este servidor Windows Server se instalara el manejador de base de datos SQL Server, pero no será parte del manual de instalación del proyecto ya que se encuentra instalado en el server.

8.4.7. Configuración de Servidores Linux.

Luego a continuación continuaremos con la configuración de los servidores espejo de sistema operativo centos (Cluster de Servidores), recuerden que ya hemos trabajado el equipo Windows con discos espejo y tarjetas de red con balanceo de red.

Instalaremos el software Conga, que es un conjunto integrado de componentes de software que proporciona tareas de configuración y administración centralizada para los cluster, entre ellos está la aplicación luci y ricci.

8.4.7.1. Instalación de Cluster de Servidores

Comencemos la instalación del software para Clúster de servidores, para mejor administración de los dos servidores de aplicaciones y archivos se realizara la creación de una máquina virtual en el servidor Windows con el sistema operativo centos de usos exclusivo solo de administración de cluster. Recordar que esta máquina virtual también se agrega en el servicio DNS como un host nuevo. A este equipo lo llamaremos virtual1.

Estos pasos lo hacemos en los dos nodos de linux nodoapp1 y nodoapp2 y la máquina virtual.

```
# yum install luci ricci -y
# yum groupinstall "High Availability" -y
```

Seteamos la clave de ricci a redhat en los 3 servers

```
# passwd ricci
    Changing password for user ricci.
    New password:
    BAD PASSWORD: it is based on a dictionary word
    BAD PASSWORD: is too simple
    Retype new password:
    passwd: all authentication tokens updated successfully.
```

Iniciamos luci en el virtual1 que es nuestro nodo de administracion

```
# service luci start
```

```
# chkconfig luci on
```

Iniciamos ricci en los 3 nodos que son nuestro nodos agentes de clustering (nodoapp1, nodoaap2 y virtual1)

```
# service ricci start
# chkconfig ricci on
```

Abrimos un explorador de internet y colocamos la direccion de luci en el virtual1 para configurar el cluster via web

```
https://192.168.0.111:8084
```

Aceptamos la excepciones de seguridad para acceder al web y nos logueamos con las credenciales de root

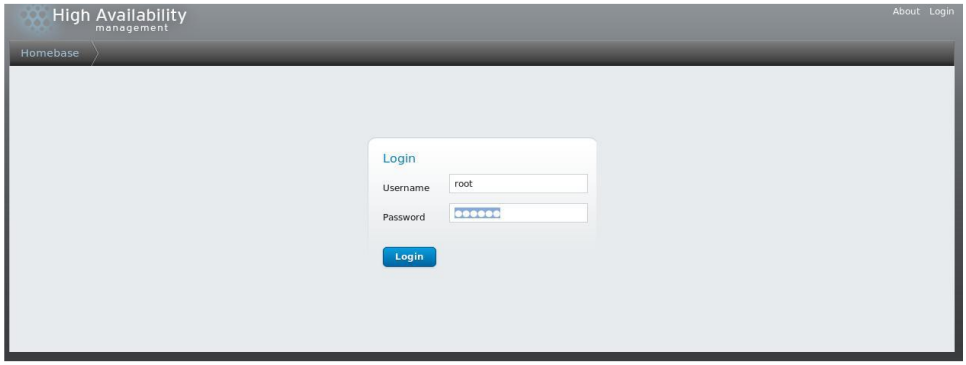

Ilustración 42 Credenciales de acceso

Luego creamos un nuevo cluster en la opción Manager_Clusters

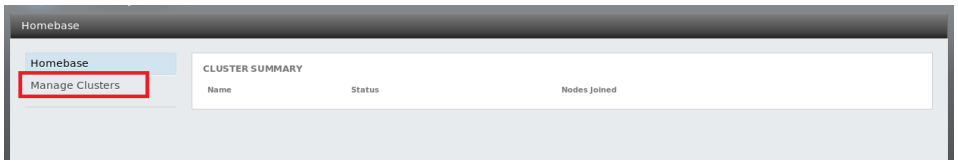

Ilustración 43 Acceso a laadministracion del cluster

Luego damos click en la opción CREATE

Ilustración 44 Crear Cluster

Luego en la ventana adicional debemos agregar ya los nodos que conformaran el cluster, en nuestro caso virtual1, nodoaap1 y nodoapp2.

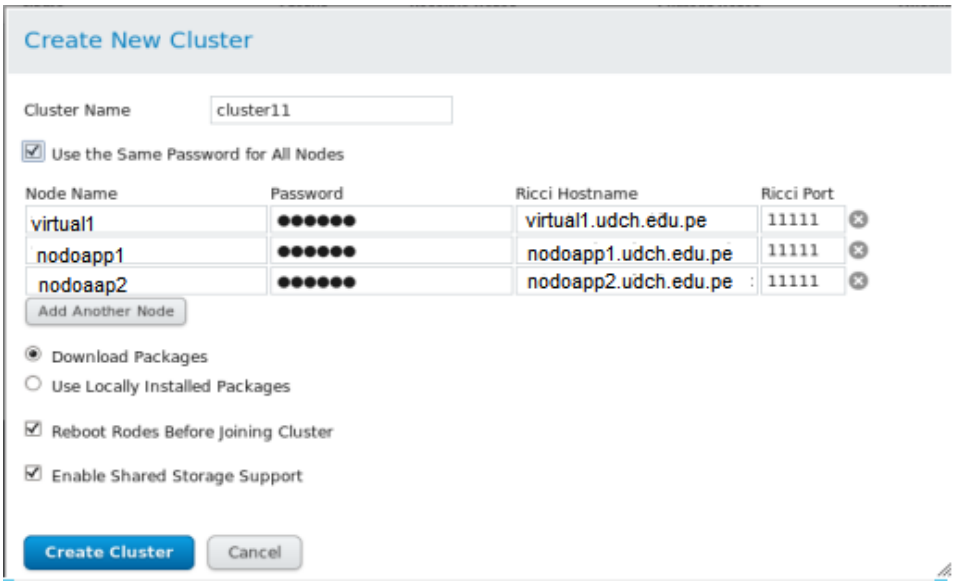

Ilustración 45 Nodos del Cluster

Luego damos en la opción Create cluster y listo.

Procedimiento para crear un balanceador de carga del servidor de aplicaciones (apache con lectura de archivos php).
Instalamos el software de balanceo dentro de virtual1

```
[root@virtual1 ~]# yum install piranha ipvsadm –y
```

Iniciamos y habilitamos los servicios

```
[root@virtual1 ~]# service piranha-gui start
[root@virtual1 ~]# chkconfig piranha-gui on
[root@virtual1 ~]# chkconfig pulse on
```

Creamos una clave para piranha

```
[root@virtual1 ~]# piranha-passwd
New Password:
Verify:
Updating password for user piranha
```

Modificamos la línea en /etc/sysctl.conf

```
net.ipv4.ip_forward = 1
# sysctl -p
```

Ingresamos a la interfaz web de piranha con la url

http://192.168.0.111:3636

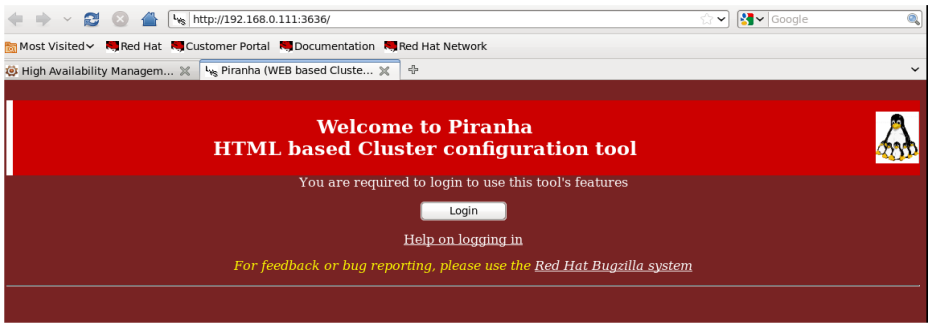

Ilustración 46 Interfaz de Piranha

Luego procedemos a autenticarnos con el usuario y clave para inciar la configuración del balanceo de carga.

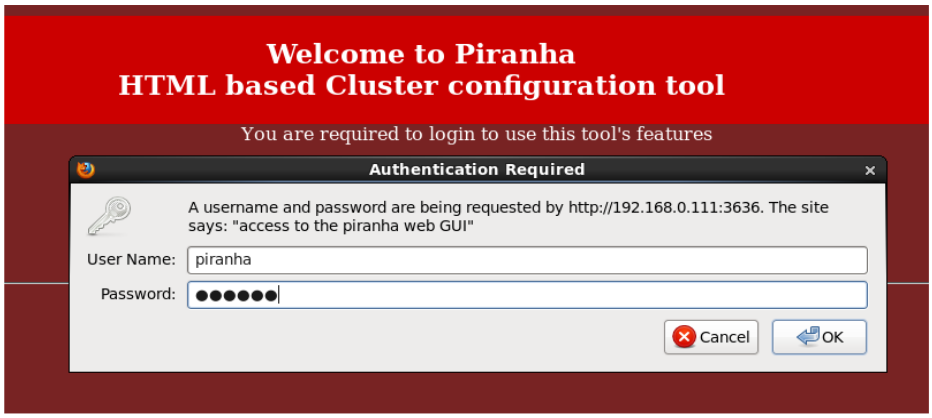

Ilustración 47 Credenciales de acceso a Piranha

Luego ingresamos los datos Ip del servidor virtual1 que es el que administrará el balanceo de carga del cluster.

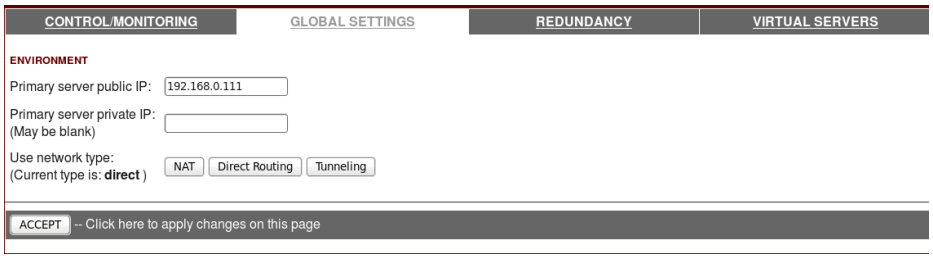

Ilustración 48 Datos del Servidor balanceador de carga

Luego ingresaremos datos de nuestro equipo virtual balanceador (no confundir con virtual1, que es el que administra el cluster), es decir el ip que se accederá desde la web y que englobara a ambos servidores de aplicaciones reales (nodoapp1 y nodoaap2). Solo se ingresara un ip del rango de la red local y será usado para este fin de balanceador y acceso a la página web. Para nuestro desarrollo implementaremos el ip 192.168.0.61/32.

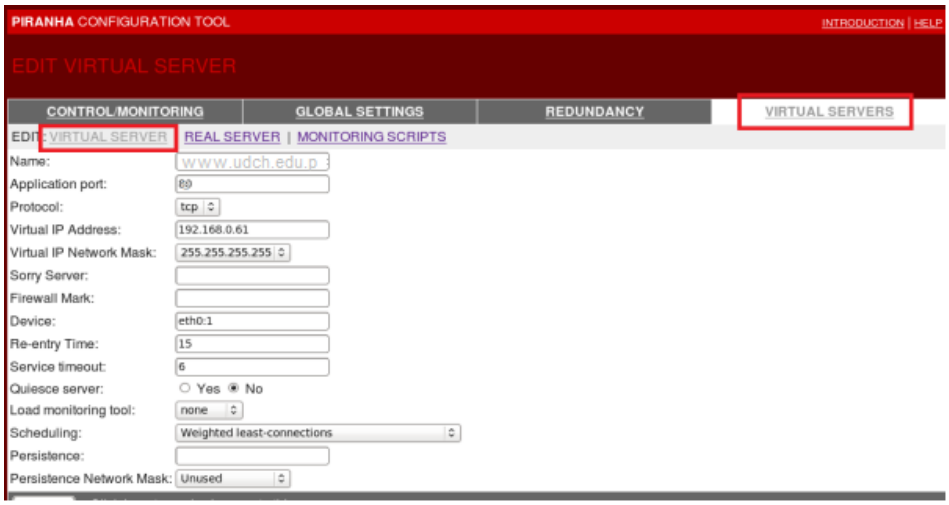

Ilustración 49 Configuracion del virtual balanceador

Finalmente adicionamos los dos nodos reales al balanceador de carga de servidor de aplicaciones.

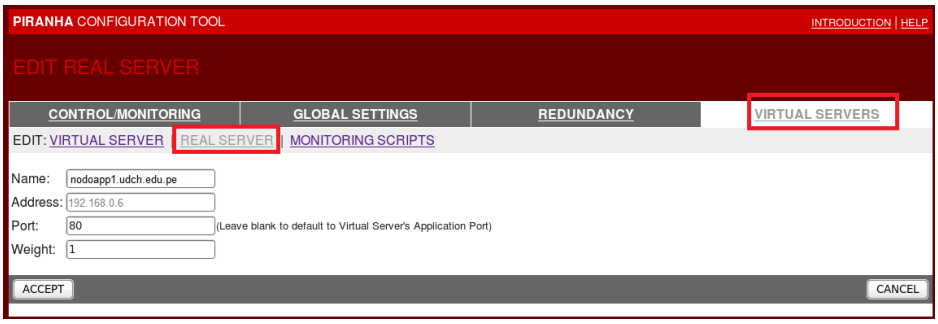

Ilustración 50 Adicionar nodos reales a balancear

Una vez configurado procederemos a instalar las aplicaciones apache con soporte a php y su configuración para acceso a una base de datos mysql. Esto lo haremos en los dos nodos de aplicaciones

```
#yum –y install httpd php
```

Y dentro de la carpeta /var/www/html copiamos la aplicación con soporte a php y conexión a una base de datos. Este paso lo obviaremos ya que no es parte del proyecto.

Finalmente para que ambos nodos reconozca el ip de la máquina virtual de balanceo de servidor de aplicaciones agregamos las siguientes líneas.

En ambos servidores appnodo1 y appnodo2 agregamos la interfaz virtual

```
# ip addr add 192.168.0.61 dev eth0
```

Agregamos la línea en el archivo /etc/rc.local para que cada vez que reinicie los nodos reales se cargue automáticamente esta interface.

```
/sbin/ip addr add 192.168.0.61 dev eth0
```

No olvidar copiar las aplicaciones web en la carpeta del apache y luego cargar en el navegador http://www.udch.edu.pe y deberá aparecer la aplicación y para realizar la prueba se deberá de ir

deteniendo los servicios en los nodos y probar el funcionamiento de la aplicación.

8.4.7.2. Montar cluster Mysql.

Para instalara nuestro Cluster de Mysql hay que tener en cuenta la siguiente arquitectura:

- 1 nodo de administración (MGM), este nodo se instalara como máquina virtual en la Pc Windows (Windows server 2012)
- 2 nodos de datos (NDB) que a su vez van a ser nodos SQL (MYSQLD) instalados en los servidores Linux.

Gráficamente se muestra el diseño lógico en la figura siguiente:

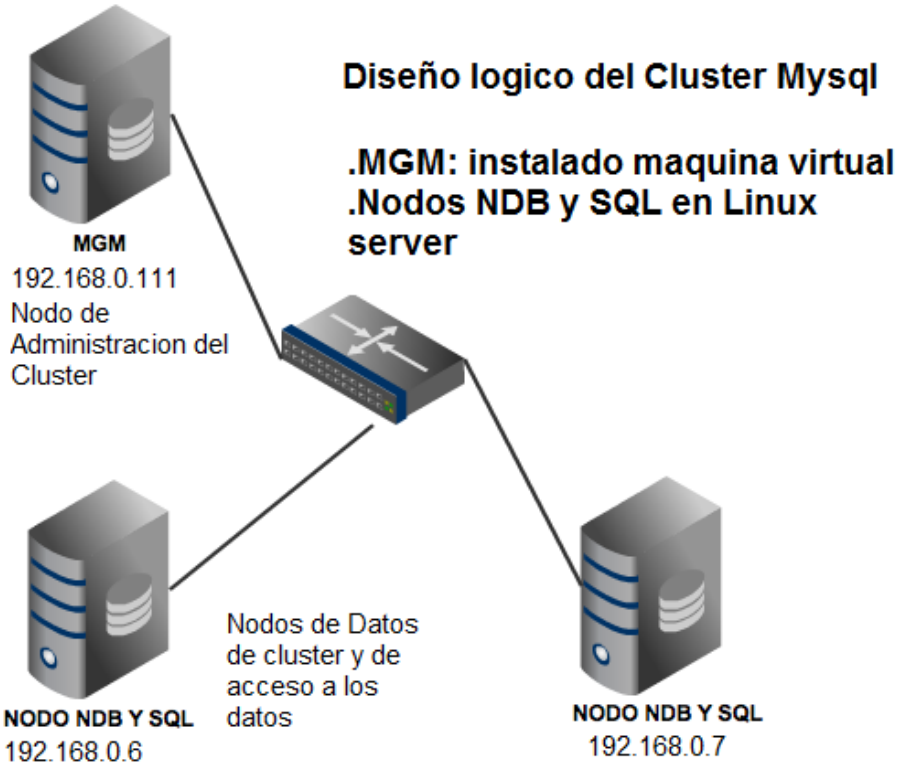

Ilustración 51 Diseño logico del cluster Mysql

Una vez entendido la topología lógica a configurar lo que tenemos que hacer es configurar el nodo de administración o MGM. Recordar que el nodo de administración del cluster está en la máquina virtual en Windows server y procedemos a instalar la aplicación MySQL con el soporte de NDBCluster.

Para comenzar con el procedimiento de instalación, descargue el clúster Mysql NDB 7.4.10 con el comando wget.

```
# wget http://dev.mysql.com/get/Downloads/MySQL-Cluster-7.4/MySQL-Cluster-gpl-7.4.10-1.el6.x86_64.rpm-bundle.tar
```

Descargue el paquete perl-data-dumper usando el comando wget.

```
# wget ftp://195.220.108.108/linux/dag/redhat/el6/en/x86_64/dag/RPMS/perl-Data-Dumper-2.125-1.el6.rf.x86_64.rpm
```

Elimine los archivos de la biblioteca MySQL ejecutando el siguiente comando.

```
[ root @virtual1 ~] # yum eliminar mysql-libs y ndash
```

A continuación, instale el paquete de volcado de datos perl ejecutando el siguiente comando.

```
# yum install perl-Data-Dumper-2.125-1.el6.rf.x86_64.rpm
```

Ahora, extraiga el paquete de clúster MySQL NDB ejecutando el comando tar.

```
# tar -xvf MySQL-Cluster-gpl-7.4.10-1.el6.x86_64.rpm-bundle.tar
```

Instale los siguientes paquetes MySQL NDB de la siguiente manera.

```
# yum instalar MySQL-Cluster-shared-gpl-7.4.10-1.el6.x86_64.rpm MySQL-Cluster-server-gpl-7.4.10-1.el6.x86_64.rpm MySQL-Cluster -client-gpl-7.4.10-1.el6.x86_64.rpm & ndash
```

Recordar que estos pasos se realizan en los tres nodos del cluster Mysql.

Ahora cree un directorio en la ubicación a continuación ejecutando los siguientes comandos.

```
[ root @virtual1 ~] # mkdir / var / lib / mysql-cluster
[ root @virtual1 ~] # cd / var / lib / mysql-cluster
```

A continuación, cree un archivo en el directorio recién creado y agregue los detalles de configuración del clúster al archivo. Guardar y salir del archivo.

```
[ root @virtual1 mysql-cluster] # vim config.ini
[ndb_mgmd]
# Nodo de gestión db1
HostName = 192.168.0.111

[ndbd predeterminado]
NoOfReplicas = 2 # Número de réplicas
DataMemory = 100M # Asignación de memoria para almacenamiento de datos
IndexMemory = 100M # Asignación de memoria para almacenamiento de índice

#Directorio para el nodo de datos
DataDir = / var / lib / mysql-cluster

[ndbd]
#Data Node db2
HostName = 192.168.0.6

[ndbd]
#Data Node db3
HostName = 192.168.0.7

[mysqld]
HostName = 192.168.0.6 #SQL Node1
```

```
[mysqld]
HostName = 192.168.0.7 #SQL Node2
```

Inicie el nodo de administración NDB ejecutando el siguiente comando.

```
[ root @ virtual1 ~] # ndb_mgmd --config-file = / var / lib / mysql-cluster / config.ini
MySQL Cluster Management Server mysql-5.6.28 ndb-7.4.10
2017-06-09 01:34:57 [MgmtSrvr] INFORMACIÓN - El directorio de configuración predeterminado '/ usr / mysql-cluster' no existe. Intentando crearlo …
2017-06-09 01:34:57 [MgmtSrvr] INFORMACIÓN - Directorio de configuración creado con éxito.
```

Verifique el estado del otro nodo desde el nodo de administración NDB .

```
- Clúster NDB - Cliente de gestión -
 ndb_mgm> show
Conectado a Management Server en: localhost: 1186
Configuración de clúster
---------------------
[ndbd (NDB)] 2 nodo (s)
id = 2 (no conectado, aceptando conectarse desde 192.168.0.6)
id = 3 (no conectado, aceptando conectarse desde 192.168.0.7)

[ndb_mgmd (MGM)] 1 nodo (s)
id = 1 @ 192.168.0.111 (mysql-5.6.28 ndb-7.4.10)

[mysqld (API)] 2 nodo (s)
```

id = 4 (no conectado, aceptando conectarse desde 192.168.0.6)
id = 5 (no conectado, aceptando conectarse desde 192.168.0.7)
ndb_mgm> salir

Cambie al nodo de nodoapp1 y realice los siguientes pasos. Cree el nuevo directorio para los datos de la base de datos que se definen en el nodo de administración mediante el archivo de configuración " config.ini ".

```
[ raíz @ nopdpapp1 ~] # mkdir -p / var / lib / mysql-cluster
```

Luego, configure el nodo de datos NDB .

```
[ root @ nodoapp1 ~] # vim /etc/my.cnf
[mysqld]
ndbcluster
ndb-connectionstring = 192.168.0.111
[mysql_cluster]
ndb-connectionstring = 192.168.0.111
```

Ahora inicie el nodo de datos siguiendo el comando.

```
[ raíz @ nodoapp1 ~] # ndbd
2017-06-09 03:48:45 [ndbd] INFORMACIÓN - user conectado a '192.168.0.6:1186'
2017-06-09 03:48:45 [ndbd] INFORMACIÓN - user asignó nodeid: 2
```

Realizar los pasos anteriores para el nodo de datos 02 es decir para nodoapp2
Ahora configuramos los nodos SQL que son los mismos equipos que los NDB (nodoapp1 y nodoapp2) realizamos la siguiente configuración.

```
[ root @ nodoapp1 ~] # vim .my.cnf
 [mysqld]
ndbcluster
```

```
ndb-connectionstring = 192.168.0.111 # Dirección IP para el nodo de administración del servidor
default_storage_engine = ndbcluster # Definir el motor de almacenamiento predeterminado utilizado por MySQL

[mysql_cluster]
ndb-connectionstring = 192.168.0.111 # Dirección IP para el nodo de administración del servidor
```

Establezca las credenciales de MySQL ejecutando los siguientes comandos.

```
[ root @nodoapp1 ~] # mysql_secure_installation
 [ root @ nodoapp1 ~] # service mysql restart
```

Recuerde repetir los mismos pasos para el nodo 2 de datos es decir para nodoapp2.

Cambie al nodo de administración (virtual1) y verifique el estado de la conexión del clúster.

```
[ raíz @virtual1 ~] # ndb_mgm
- Clúster NDB - Cliente de gestión -

ndb_mgm> show
Conectado a Management Server en: localhost: 1186
Configuración de clúster
---------------------
[ndbd (NDB)] 2 nodo (s)
id = 2 @ 192.168.0.6 (mysql-5.6.28 ndb-7.4.10, Nodegroup: 0, *)
id = 3 @ 192.168.0.7 (mysql-5.6.28 ndb-7.4.10, Nodegroup: 0)

[ndb_mgmd (MGM)] 1 nodo (s)
id = 1 @ 192.168.0.111 (mysql-5.6.28 ndb-7.4.10)
```

```
[mysqld (API)] 2 nodo (s)
id = 4 @ 192.168.0.6 (mysql-5.6.28 ndb-7.4.10)
id = 5 @ 192.168.0.7 (mysql-5.6.28 ndb-7.4.10)
ndb_mgm> quit
```

Ahora es el momento de verificar el estado de replicación. En el nodoapp1 de SQL, cree una base de datos ejecutando el siguiente comando.

```
mysql> créate database cluster1_db
 Consulta OK, 1 fila afectada (0.19 segundos)

mysql> show databases
+ ----------------------------------+
| Database                          |
+ ----------------                  +
|      información_esquema          |
|      cluster1_db                  |
|      mysql                        |
|      ndbinfo                      |
|      performance_schema           |
+ ------------------------          +
```

Cambie a SQL nodeapp2 y verifique el estado de las bases de datos.

```
mysql> show databases
+ ----------------------------------+
| Database                          |
+ ----------------                  +
|      información_esquema          |
|      cluster1_db                  |
```

```
| mysql                    |
| ndbinfo                  |
| performance_schema       |
+ ------------------------- +
```

Ahora cree una base de datos en SQL nodeapp2 para verificar el proceso de replicación.

```
mysql> créate database cluster2_db
Consulta OK, 1 fila afectada (0.07 segundos)
mysql> show databases
+ ------------------------------------+
| Database                           |
+ ------------------                 +
|      información_esquema           |
|      cluster1_db                   |
|      cluster2_db                   |
|      mysql                         |
|      ndbinfo                       |
|      performance_schema            |
+ ------------------------------  +
```

Cambie al nodoapp1 de SQL y verifique el proceso de replicación.

```
mysql> show databases
+ ------------------------------------+
| Database                           |
+ ------------------                 +
|      información_esquema           |
|      cluster1_db                   |
|      cluster2_db                   |
|      mysql                         |
|      ndbinfo                       |
|      performance_schema            |
```

```
+ ---------------------------------          +
```

Con estos pasos terminamos la configuración de Cluster de Mysql Server.

8.4.7.3. Crear Cluster de Carpetas compartidas

Configurar la resolución de nombres en ambos nodos (nodoapp1 y nodoapp2)

```
#vim /etc/hosts
192.168.0.6 nodoapp1
192.168.1.7 nodoapp2
```

Instalar DRBD en ambos nodos

```
#rpm -ivh http://www.elrepo.org/elrepo-release-7.0-2.el7.elrepo.noarch.rpm
#rpm --import /etc/pki/rpm-gpg/RPM-GPG-KEY-elrepo.org
#yum --skip-broken -y install drbd*-utils kmod-drbd*
```

Configurar la carga del módulo en ambos nodos

```
#vim /etc/modules-load.d/drbd.conf
#reboot
```

Verificar la carga del módulo DRDB en ambos nodos

```
#lsmod | grep drbd
 Resultado:
drbd 405376 0
libcrc32c 12644 1 drbd
```

Crear la partición en ambos nodos

#fdisk /dev/sda

Command (m for help): n

Command action

e extended

p primary partition (1-4)

p

Partition number (3,4, default 3): [Enter]

First sector (12584960-20971519, default 12584960): [Enter]

Last sector, +sectors or +size{K,M,G} (12584960-20971519, default 20971519): [Enter]

Using default value 20971519

Partition 3 of type Linux and of size 4 GiB is set

Command (m for help): w

Reiniciar

```
# reboot
```

Formatear la nueva partición en ambos nodos

```
#mke2fs -j /dev/sda3
#dd if=/dev/zero bs=1M count=500 of=/dev/sda3; sync
```

Configurar DRDB en el nodoapp1

```
#vi /etc/drbd.d/testdata1.res
resource testdata1 {
protocol C;
on nodoapp1.local {
 device /dev/drbd0;
 disk /dev/sda3;
 address 192.168.0.6:7788;
 meta-disk internal;
 }
on nodoapp2.local {
 device /dev/drbd0;
```

```
 disk /dev/sda3;
 address 192.168.0.7:7788;
 meta-disk internal;
 }
}
```

Copiar el archivo testdata1.res al nodoapp2

```
#scp /etc/drbd.d/testdata1.res root@192.168.0.7:/etc/drbd.d/
```

Iniciar DRDB META DATA STORAGE en ambos nodos

```
[root@nodoapp1 ~]# drbdadm create-md testdata1
[root@nodoapp2 ~]# drbdadm create-md testdata1
```

Iniciar el servicio de DRDB en ambos nodos

```
#systemctl start drbd
#systemctl enable drbd
```

Configurar el nodoapp1 como DRDB primario

```
#drbdadm primary testdata1
#drbdadm primary testdata1 --force
```

Formatear la partición DRBD

```
[root@nodoapp1 ~]# mkfs.ext4 /dev/drbd0
```

Probar funcionamiento de la partición DRBD

Crear un archivo en la partición DRDB

```
[root@nodoapp1 ~]# mount /dev/drbd0 /mnt
[root@nodoapp1 ~]# touch /mnt/prueba1.txt
[root@nodoapp1 ~]# ls /mnt/
[root@nodoapp1 ~]# umount /mnt
```

Hacer que el nodoapp1 sea el secundario

```
[root@nodoapp1 ~] drbdadm secondary testdata1
```

Hacer que el nodoapp2 sea el primario, luego montar la partición y verificar el archivo prueba2.txt

```
[root@nodoapp2 ~] drbdadm primary testdata1
[root@nodoapp2 ~] mount /dev/drbd0 /mnt
[root@nodoapp2 ~] ls /mnt
[root@nodoapp2 ~] umount /mnt
```

Hacer que el nodoapp2 sea secundario nuevamente

```
[root@nodoapp2 ~] drbdadm secondary testdata1
```

Hacer que el nodoapp1 sea primario nuevamente

```
[root@nodoapp1 ~] drbdadm primary testdata1
```

Verificar DRDB

```
#drbd-overview
```

Crear el CLUSTER para automatizar alta disponibilidad instalando la aplicación Corosync y Pacemaker en ambos servidores

```
#yum -y install corosync pcs pacemaker
```

Cambiar la contraseña del usuario hacluster en ambos nodos

```
#passwd hacluster
Password:
```

Iniciar el servicio PCS en ambos nodos

```
#systemctl start pcsd
```

Configurar el inicio de los servicios en ambos nodos

```
#systemctl enable pcsd.service
#systemctl start pcsd.service
#systemctl enable corosync.service
```

```
#systemctl enable pacemaker.service
```

Configurar y autenticar los nodos, realizar solo en el nodoapp1

```
#pcs cluster auth nodoapp1 nodoapp2
Username: hacluster
Password:
```

Crear el cluster y configurar parámetros, realizar solo en el nodo1

```
#pcs cluster setup --name cluster_voip nodo1.local nodo2.local
#pcs cluster start --all
#pcs cluster enable --all
#pcs property set stonith-enabled=false
#pcs property set no-quorum-policy=ignore
```

Crear recurso virtual_ip para IP Flotante

```
#pcs resource create virtual_ip ocf:heartbeat:IPaddr2 ip=192.168.1.15 cidr_netmask=32 op monitor interval=30s on-fail=restart
#pcs cluster cib drbd_cfg
#pcs cluster cib-push drbd_cfg
```

Crear recurso DRBD y crear la replicación para los nodos

```
#pcs -f drbd_cfg resource create DrbdData ocf:linbit:drbd drbd_resource=testdata1 op monitor interval=60s
#pcs -f drbd_cfg resource master DrbdDataClone DrbdData master-max=1 master-node-max=1 clone-max=2 clone-node-max=1 notify=true
#pcs status resources
#pcs cluster cib-push drbd_cfg
#pcs status resources
```

Crear recurso FILESYSTEM para el punto de montaje automatizado

```
#pcs cluster cib fs_cfg
#pcs -f fs_cfg resource create DrbdFS Filesystem device="/dev/drbd0" directory="/mnt/" fstype="ext4"
#pcs -f fs_cfg constraint colocation add DrbdFS with DrbdDataClone INFINITY with-rsc-role=Master
#pcs -f fs_cfg constraint order promote DrbdDataClone then start DrbdFS
#pcs -f fs_cfg constraint colocation add DrbdFS with virtual_ip INFINITY
#pcs -f fs_cfg constraint order virtual_ip then DrbdFS
#pcs cluster cib-push fs_cfg
#pcs status resources
```

Verificar que el punto de montaje está montado en /mnt/

```
#df –h
```

Para probar el funcionamiento del CLUSTER apagar el nodoapp1 y el nodoapp2 deberá pasar la IP virtual y el punto de montaje /mnt/ al nodoapp2. Luego apagar el nodoapp2 y todo el control pasara al nodoapp1 volviendo el punto de montaje a este de manera automática.

Para ver los logs de funcionamiento, visualizar con: tail -f /var/log/messages.

8.5. Presupuesto

Recursos y presupuestos

- **Equipo de desarrollo**

ESPECIALISTA	TOTAL

Asesor	S/1,500.00
Responsable de Implementación	S/.1,500.00
Total	S/.3,000.00

Tabla 15: Costos equipo de desarrollo

- **Documentación**

DESCRIPCION	**CANT.**	**TOTAL**
FOTOCOPIAS	600	S/.200.00
IMPRESIONES	450	S/.300.00
ESPIRALADOS	5	S/. 100.00
Empastados	4	S/.200.00
Total		S/.800.00

Tabla 16 Costos Documentación

- **Otros**

DESCRIPCION	**CANT**	**PRECIO UNI.**	**TOTAL**
PASAJES	--	--	S/.500.00
PAQUETE DE DATOS	--	--	S/.150.00
TOTAL			S/.650.00

Tabla 17: Costos Otros

Resumen del Presupuesto:

- Equipo de desarrollo : S/. 3,000.00
- Documentación : S/. 800.00
- Otros : S/. 650.00

TOTAL	:	S/.4,450.00

Recuperación.

Se considera que el ahorro en tecnología convencional equivalea S/. 890.00en forma mensual por lo tanto la recuperación se logra en 06 meses calendario.

Capítulo IX:
REFERENCIAS BIBLIOGRÁFICAS

CAPITULO IX: BIBLIOGRAFIA

- Andrew S. (2013). Organización de computadoras. Un enfoque estructurado. Tanenbaum. 4Ta Edición. Ed. Pearson Educacion.
- Douglas E. Comer - Redes Globales de Información con Internet y TCP/IP (2014). Tercera Edición. PRENTICEL-HALL HISPANOAMERICANA, S.A
- FITZGERALD, DENNIS(2003) - Redes y Comunicación de datos en los negocios. Editorial Limusa, S. A. de C. V.
- Landívar, Edgar (2008) - Comunicaciones Unificadas con Elastix. Primera Edición 2008. Edición virtual.
- Lara, E. (2013) .Introducción a las redes con Linux Server. (En línea) Consultado el 10 de agosto. Formato PDF. Disponible en: http://personals.ac.upc.edu/elara/documentacion/LINUX%20%20UD1%20-%20Introduccion%20Linux.pdf
- Leon-Garcia, Alberto (2010). REDES DE COMUNICACIÓN Conceptos fundamentales y arquitecturas básicas. MCGRAW-HILL / INTERAMERICANA DE ESPAÑA, S.A.
- LOES (Ley Orgánica de Educación superior). 2010. Función Ejecutiva. (En línea). Consultado el 20 Sept.. Formato PDF. Disponible en http://uide.edu.ec/SITE/norma_juridica.pdf
- López, Vicente (2001) - Linux Guía de Instalación y Administración. MCGRAW-HILL / INTERAMERICANA DE ESPAÑA, S.A.
- LTSP – Homepage (2014). (En Línea). Consultado el 20 setp. Formato HTML. Disponible en: http://www.ltsp.org/
- Moya, J. M. (2015). Servicios y redes de telecomunicaciones. España: Thomson Paraninfo.
- Mclver, A. y Flynn, I. (2010). Sistemas Operativos. 6ta Ed. DF, MX. p 2.
- Molina, F. (2014). Redes Locales. Ciclos Formativos Grado Medio. 1 ed. Madrid, ES.
- Moya, J, (2006). TELEFONÍA IP Y TECNOLOGÍA VoIP. Editorial: Creaciones Copyright

- Serrat , Manuel (2009). Ubuntu Linux. Olmos. Ed. RA-MA.
- Schroder, Carla (2010). Redes en Linux. Madrid, España Ed. Anaya Multimedia.
- Turnbull, James (2019). Administración de sistemas Linux. Anaya Multimedia.
- https://libuntu.wordpress.com/2015/05/16/centos-7-ahora-soporta-la-arquitectura-amr64-aarch64/ Recuperado el 25 de septiembre de 2018
- http://www.areatecnologia.com/informatica/que-es-dns.html . Visitado el 25 de septiembre de 2018
- http://www.itbuy.com.ar/NotasdeInter%C3%A9s/TabId/96/ArtMID/442/ArticleID/26/Que-son-los-DNS-.aspx. Revisado el 12 de septeimbre de 2018
- https://administracionsistemasoperativos201415.wordpress.com/2015/02/27/active-directory/ . Revisado el 12 de septiembre de 2018
- http://suncoremicrosystem.com/welcome/technology/8 Visitado el 13 de septiembre de 2018
- https://goodlogo.com/extended.info/apache-software-foundation-logo-3074 . Visitado el 20 de septiembre de 2018
- http://www.redessil.com/instalacion-de-servidor-de-archivos/ Visitado el 20 de septiembre de 2018.
- Gil, J. G. (19 de Agosto de 2018). *openwebinars*. Recuperado el 1 de setiembre de 2019, de https://openwebinars.net/blog/caracteristicas-importantes-de-postgresql/
- Gustavo. (13 de Mayo de 2019). Recuperado el 01 de setiembre de 2019, de https://www.hostinger.es/tutoriales/que-es-mysql/
- Moya, J. M. (2015). *Servicios y redes de telecomunicaciones.* España: Thomson Paraninfo.
- Rouse, M. (enero de 2015). *searchdatacenter*. Recuperado el 1 de setiembre de 2019, de https://searchdatacenter.techtarget.com/es/definicion/DB2