UNIVERSIDAD CATÓLICA ANDRÉS BELLO

FACULTAD DE INGENIERÍA

ESCUELA DE INGENIERÍA EN TELECOMUNICACIONES

“Implementación de un sistema que permita consultar el registro de usuarios de una empresa de telefonía móvil”

TRABAJO ESPECIAL DE GRADO

Presentado ante la

UNIVERSIDAD CATÓLICA ANDRÉS BELLO

Como parte de los requisitos para optar al título de

INGENIERO EN TELECOMUNICACIONES

REALIZADO POR:

Br. Battaglini Segura, Juan Carlos

TUTOR:

Freites, Jhoan

Caracas, enero del 2020

Implementación de un sistema que permita consultar el registro de usuarios de una empresa de telefonía móvil

REALIZADO POR: Battaglini Segura, Juan Carlos

TUTOR: Ing. Jhoan, Freites

Fecha: Caracas, enero del 2020

RESUMEN

El objetivo de este Trabajo de Grado es diseñar una solución para poder manipular la base de datos de todos los abonados dentro de la corporación Digitel y que dicho sistema sea capaz de ofrecer una alta fiabilidad del servicio a través de la implementación de un clúster de alta disponibilidad. Este proyecto nace por la necesidad de presentar una solución alterna, la cual ofrezca tiempos menores de respuestas ante consultas o modificaciones masivas, esto debido a que actualmente poseen una herramienta llamada PGW Web LMT, la cual requiere de mucho tiempo para llevar a cabo las actividades mencionadas de forma masiva.

Para esto se llevó a cabo una investigación para definir las etapas necesarias para concretar el objetivo establecido, estas están estrechamente relacionadas con las investigaciones bibliográficas, diseño e implementación, equipos a usar, infraestructura, instalación y evaluación del desempeño del servicio en la red, al igual que el desarrollo del tomo de TEG.

Por último, se realizó una demostración con el proyecto ya implementado, evidenciando la factibilidad y eficiencia de la plataforma diseñada a través de un listado masivo de un lote de abonados.

Dedicatoria

Agradecimientos

Índice General

Índice de figuras

Índice de Tablas

Introducción

A lo largo de los años con la globalización, las telecomunicaciones han sido factor clave para conectar a las personas entre sí, permitiéndoles rápido acceso a la información y a nuevas tecnologías, abriendo camino a un sinfín de usos para la misma en nuestra sociedad que desesperadamente busca cada vez estar más conectada entre sí, rompiendo distancias y fronteras alrededor del globo causando un impacto positivo en lo social, político, económico y cultural.

Actualmente, la Corporación Digitel cuenta con un Single SDB el cual es una base de datos centralizada en la que se encuentran almacenados los perfiles de sus subscriptores, en dichos perfiles se aprecian los servicios que tienen aprovisionados cada uno de estos usuarios de forma detallada. El objetivo, es crear una solución alterna a la disponible actualmente, reduciendo los tiempos para efectuar modificaciones o consultas masivas a dichos perfiles por parte del departamento de Operación y Mantenimiento (O&M), brindando así una optimización significativa.

Dicha optimización se pretende hacer efectiva mediante la ejecución de scripts de procesamiento de texto en lenguaje informático Perl a través de dos servidores Sun Netra X4270 con un Sistema Operativo (S.O.) SUSE Linux Enterprise Server 15.1 y la extensión Suse Linux Enterprise Server High Availability Extensión 15.1 para la implementación de un clúster de alta disponibilidad con la meta de garantizar una mejora en la disponibilidad del servicio.

Capítulo I

Planteamiento del proyecto

En el siguiente capítulo se describen las razones que dieron motivo a la investigación, comenzando desde el planteamiento del problema, hasta la defensa del motivo que da origen a este proyecto. Se plantea el objetivo general y los objetivos específicos que se pretenden completar, en conjunto a las limitaciones, alcances y contratiempos que surgieron a lo largo de este trabajo especial de grado.

I.1 Planteamiento del problema

La Corporación Digitel es una empresa de telefonía que se dedica a prestar servicios en todo el país, por ello es fundamental contar con información de sus usuarios. Tal información es mantenida por el departamento de Operación y Mantenimiento (O&M) Red Central utilizando un aplicativo llamado PGW WEM LMT. Con este aplicativo la empresa puede hacer consultas y modificaciones de los perfiles de sus abonados.

Este aplicativo PGW WEB LMT fue desarrollado por HUAWEI proveedora de los equipos para satisfacer los requerimientos de información por parte de Digitel respecto a sus abonados. Ahora bien, el crecimiento de la empresa en cuanto a cantidad de abonados hace que las consultas o modificaciones masivas sean muy lentas utilizando esta herramienta, con lo cual esto presenta un problema de eficiencia para la empresa en cuanto a la modificación de los servicios ofrecidos o solicitados por sus abonados.

I.1.1 Solución propuesta

Como solución se plantea una nueva aplicación diseñada y desarrollada de acuerdo a los requerimientos de la Corporación Digitel. Esta aplicación se desarrollará utilizando el lenguaje de programación Perl, debido a requerimientos de la empresa, y para garantizar la alta disponibilidad de la misma se instaló en un clúster de servidores.

I.2 Objetivos de la Investigación

A continuación, se explica detalladamente los objetivos del presente Trabajo especial de Grado.

I.2.1 Objetivo General

Implementar un sistema que permita reducir los tiempos de consulta de usuarios registrados a los servicios móviles de Digitel, utilizando un servidor en clúster para garantizar alta disponibilidad

I.2.2 Objetivos Específicos

- Investigar antecedentes bibliográficos y tecnologías sobre el tema.
- Investigar la operación actual de la red DIGITEL.
- Determinar las consultas de mayor importancia para realizarlas a través de un programa desarrollado a través de un lenguaje de programación.
- Diseñar el sistema tentativo a implementar.
- Implementar sistema propuesto.
- Realizar pruebas de operatividad y desempeño.
- Redactar y presentar el tomo donde se analicen los resultados obtenidos y se presenten conclusiones y recomendaciones.

I.3 Justificación del problema

La Corporación Digitel actualmente cuenta con un sistema para efectuar consultas o modificaciones a los perfiles de usuarios de su red llamado PGW Web LMT, sin embargo, este sistema no es apto para consultas o modificaciones a gran escala, ya que los tiempos requeridos son cortos y el sistema actual no está diseñado para procesar peticiones masivas, motivo por el cual se desarrolla este trabajo especial de grado, con el objetivo de ofrecer un sistema que, en menor tiempo efectúe peticiones a gran escala.

I.4 Alcances y limitaciones

I.4.1 Alcance:

Este Trabajo de Grado tiene por alcance el diseño, desarrollo e implementación de un sistema alterno a través del cual se logre modificar o listar el registro de usuarios almacenados en el Single SDB, dicho sistema contará con su servidor espejo y a su vez la implementación de un clúster de alta disponibilidad para garantizar una mayor fiabilidad y lograr evaluar el desempeño y funcionalidad del servicio a través de pruebas con cantidades masivas de abonados dentro de las instalaciones de la Corporación DIGITEL ubicada en la región Capital.

I.4.2 Limitaciones

Este trabajo incluirá el desarrollo de las instalaciones necesarias para la implementación del servidor capaz de gestionar la información de los usuarios alocada en el Single SDB a través de un servidor con su espejo que lo respalde en las instalaciones de la corporación Digitel ubicada en la Región Capital. También se realizarán las modificaciones necesarias en el código a ejecutar para realiza las consultas o modificaciones.

El servidor implementado será de uso exclusivo la Corporación Digitel para detectar fallas o en su defecto implementar modificaciones a gran escala.

- Se utilizarán equipos que son propiedad de la empresa.
- Las facilidades donde residirá el sistema a implementar serán las aprobadas por la empresa.

Capítulo II

MARCO REFERENCIAL

II.1 Redes móviles GSM– redes móviles de 2ª generación

"Sistema desarrollado como un estándar europeo abierto y su implementación dejo resolver la itinerancia internacional (Roaming), permitiendo así que un mismo terminal móvil pueda operar con un único número de teléfono en todos los países que posean dicho sistema. Este es ampliamente utilizado en la actualidad". [1, p. 23]

Este estándar es utilizado desde principios del siglo XXI y se conoce como red 2G o de segunda generación debido a que representa la evolución de las redes analógicas a digitales.

La arquitectura de la red GSM está conformada por:

II.1.1 HLR

Registro de Posición de Inicio, *del inglés Home Location Register;* Es el elemento de la red que almacena los datos de los usuarios. Para dar de alta a un usuario en una red móvil, se deben introducir los datos en el HLR.

El HLR utilizado en la Corporación Digitel es de marca Huawei y utiliza las siguientes interfaces para para la interconexión con los otros elementos:

- Gr: para la interconexión entre SGSN (*Nodo de soporte GPRS en servicio, del inglés Serving GPRS Support Node, se encarga de dar acceso a los terminales móviles hacia la red de datos.)* y el HLR.
- C/D: para las interconexiones entre el MSS (*interruptor de software móvil, del inglés Mobile Soft Switch), se encarga de proporcionar el control de llamada, procesamiento de llamadas, entre otros.)* y el HLR.

II.1.2 EIR

Registro de Identificación del Equipo, *del inglés Equipment Identification Register*; Su labor es comprobar el identificador del dispositivo o IMEI (Identificacion Internacional del equipo móvil, de inglés International Mobile equipment identification). Todos los equipos móviles tienen un código IMEI único.

El EIR existente en la red Digitel es marca Mavenir y hace de base de datos en la cual se dispone de la información de cuales IMEI están en la lista negra (Bloqueo) y lista gris (Observación).
El EIR utiliza las siguientes interfaces para la interconexión con los otros elementos:

- F: para la interconexión entre el EIR y el MSS, sus funciones van desde el control del dispositivo móvil realizando la identificación del usuario en combinación con el HSS hasta la elección del elemento SGW *(Puerta de enlace de servicio, del inglés Serving Gateway)*, que va a gestionar la comunicación.
- GF: para la interconexión entre SGSN (*Nodo de soporte GPRS de servicio, del inglés Serving GPRS support node*) y el EIR. Es la interfaz utilizada para las consultas de chequeo de IMEI.

II.1.3 BSS

"Controlador de Estación Base, *del inglés Base Station Controller;* Varias estaciones base están asignadas a un BSC, cuya principal tarea es la de asignación y liberación de canales radio para la comunicación con los terminales móviles y garantizar que los procesos de handover funcionen correctamente" [1, p. 27]. Para que este sistema opere adecuadamente, es necesario que cada dispositivo móvil indique la celda en la que se ubica. El dispositivo móvil es capaz de detectar y monitorear la potencia de las señales provenientes de distintas estaciones bases, seleccionando aquella con mayor potencia y estabilidad para implantar conexión.

NSS – *Subsistema de Conmutación de Red, del inglés Network Switching Subsystem:* Realiza la conmutación de llamadas entre el dispositivo móvil y otros usuarios de la red GSM, así como la gestión de servicios móviles como la autenticación.

VLR – *Registro de Localización del Visitante, del inglés Visitor Location Register*: "Almacena temporalmente la información más reciente sobre la situación de un terminal móvil en el rango de su MSC (*Centro de conmutación móvil, del inglés Mobile switching center*). El VLR solicita y obtiene datos del HLR y si el terminal móvil abandona la zona visitada sus datos se eliminan del VLR" [1, p. 27].

OSS – *Subsistema de Soporte y Operación, del inglés Operation Support Subsystem:* El OSS es responsable de la operación de BSS y NSS y se encarga del mantenimiento y explotación de la red.

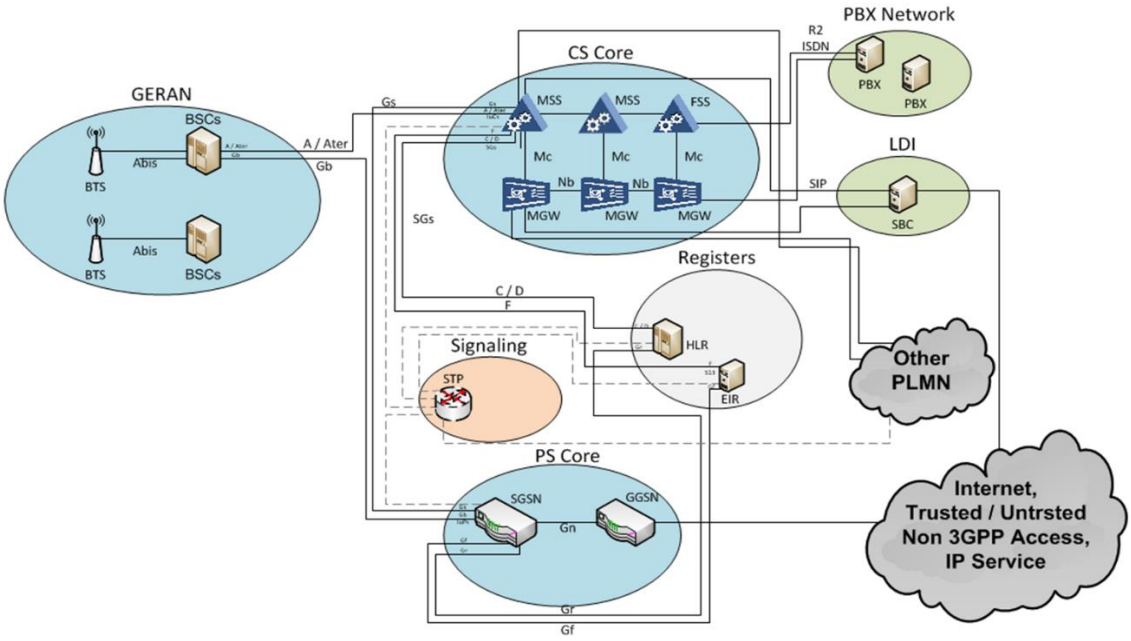


Ilustración 1. Arquitectura del Sistema GSM Red Digitel [2]

II.2 UMTS

Sistema universal móvil de telecomunicaciones – *del inglés Universal Mobile Telecommunication System*; "la segunda generación de sistemas de telecomunicaciones, tales como GSM, permite transmisión de tráfico de voz sobre entorno inalámbrico. Sin embargo, las redes 2G son incapaces de satisfacer todos los requisitos (cada vez mayores) de transmisión de datos, derivados por el rápido desarrollo de aplicaciones móviles que precisan transmisiones de datos de alta velocidad (video on demand, descarga de imágenes de alta calidad, etc.). Para cumplir estos requisitos se desarrolló la siguiente generación de redes móviles, conocida como 3G o tercera generación". [1, p. 41]

Su arquitectura se compone de tres partes esenciales, tal como es sugerido en la ilustración 2. La arquitectura de red a nivel lógico está dividida en: UE (equipo de usuario), la UTRAN (la Red Universal de Acceso Radioeléctrico Terrestre, *del inglés Universal Terrestrial Radio Access Network)*, y la CN (Red del Núcleo, *del inglés Core Network*). Como propósito principal de sus interfaces está el permitir la comunicación directa entre las distintas entidades, permitiendo su cooperación y coordinación.

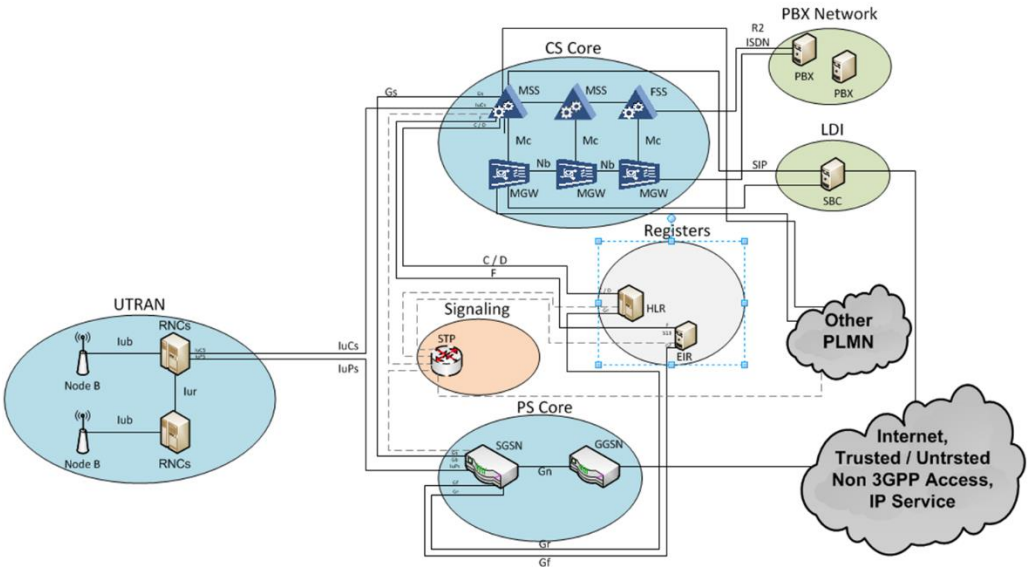


Ilustración 2. Estructura de la red UMTS Red Digitel [2]

Lo primero a considerar en la arquitectura de red UMTS es el UE, y consta de dos bloques:

MT – Terminal Móvil, *del inglés Mobile Terminal*: Dispositivo físico que gestiona todas las comunicaciones en la interfaz Uu.

La segunda parte de la arquitectura de la red UMTS está representada por la red UTRAN, consta de dos elementos de la red:

Nodo B: Es el componente responsable de la transmisión y recepción radio entre el terminal móvil y una o más celdas. Utilizando la banda de frecuencias 900 MHz en la Corporación Digitel [2]. Este elemento utiliza las siguientes interfaces para la interconexión con los otros elementos:

- Uu: Para la interconexión entre la RNC y el terminal móvil.
- IuB: para la interconexión entre el NodoB y la RNC.

RNC - *Controlador de Redes de Radio, del inglés Radio Network Controller:* es responsable de manejar el tráfico y la señalización entre un teléfono móvil y el CN (MSS, MGW- *pasarela de medios de comunicación, del inglés Media gateway*, SGSN). El RNC controla a uno o varios NodoB [2]. Este elemento utiliza las siguientes interfaces para la interconexión con los otros elementos:

- Iur: Para la interconexión entre las RNC.
- IuCs: Para la interconexión entre el MSS y la RNC. (Interfaz para la señalización)
- IuPs: Interfaz de señalización y datos entre el SGSN y la RNC.

La última sección de la red UMTS se compone por el CN, este se divide a nivel lógico en los dominios de conmutación de circuitos (CS – Circuitos Conmutados, *del inglés Circuited Switched*) y conmutación de paquetes (PS – Paquetes Conmutados, *del inglés Packet switched*). La estructura del CN es similar a la de redes 2G, razón por la cual solo se mencionarán las principales diferencias respecto a 2G [1, p. 47]:

- Como ya se ha mencionado, parte de la gestión de la movilidad en UMTS se ha trasladado de la CN a la UTRAN.
- Los aspectos de seguridad se han mejorado en UMTS, por ejemplo, se adaptaron nuevos algoritmos criptográficos más robustos. La ejecución de las funciones de cifrado se realiza en RNC en vez de en CN.
- Las funciones de tratamiento de voz se hacen en CN en lugar de BSS como ocurría en 2G.

II.3 HSDPA

"Se basa en la adopción de nuevas técnicas para mejorar significativamente la tasa de bits de datos en dirección de enlace descendente (la velocidad de datos en el enlace ascendente sigue siendo la misma). Como resultado de ello, el valor máximo teórico de la tasa de bits por celda se incrementa desde 2 Mbps hasta 14,4 Mbps. Los cambios en las redes se hacen sobre todo en UTRAN y la idea clave es mover varias radios procedimientos de gestión de recursos radio a Nodo B en lugar de RNC como en versiones anteriores de UMTS. La ventaja de este cambio es que el Nodo B está mucho "más cerca" del UE. Por lo tanto, puede ser mucho más eficaz para reaccionar teniendo en cuenta la calidad variable del canal radio". [1, p. 49]

II.4 Long Term Evolution (Advanced) – LTE(-A)

"El paso siguiente a UMTS en la evolución de las redes móviles se conoce como *Evolución a Largo Plazo, del inglés Long Term Evolution* (LTE). A diferencia de UMTS, LTE utiliza acceso OFDMA (Acceso Múltiple por División de Frecuencias Ortogonales, *del inglés Orthogonal Frequency-Division Multiple Access*) y SC-FDMA (Portadora Simple OFDMA, *del inglés Single Carrier OFDMA*) para los

enlaces descendente y ascendente respectivamente en lugar de WCDMA (Acceso Múltiple por División de Código de Banda Ancha, *del inglés Wideband Code Division Multiple Access*), utilizada en UMTS. Por lo tanto, las características de transmisión son muy diferentes si las comparamos con UMTS. Sin embargo, LTE está considerado como parte de los sistemas 3G, ya que no cumple con los requisitos definidos por la ITU para las redes 4G. El primer estándar clasificado como 4G es LTE-A (*Evolución a Largo Plazo Avanzada, del inglés Long Term Evolution - Advanced*) estandarizado en junio de 2011. Es la evolución del anterior LTE, basado en los mismos principios que ambas versiones de LTE, pero está alineada con el conjunto de requisitos definidos por la ITU y conocidos como IMT-Advanced" [1, p. 58].

La arquitectura de red GSM y UMTS dieron paso al desarrollo de la arquitectura de red LTE que, a diferencia de las otras redes, fue diseñada con el propósito de soportar únicamente la conmutación de paquetes, razón por la que su arquitectura de red no permite la conmutación de circuitos. Su red de acceso está compuesta por la EUTRAN (Evolución de la Red Universal de Acceso Radioeléctrico Terrestre, *del inglés Evolved Universal Terrestrial Radio Access Network*) y EPC (Núcleo de paquetes evolucionado, *del inglés Evolved Packet Core*) tal como se muestra en la figura 3:

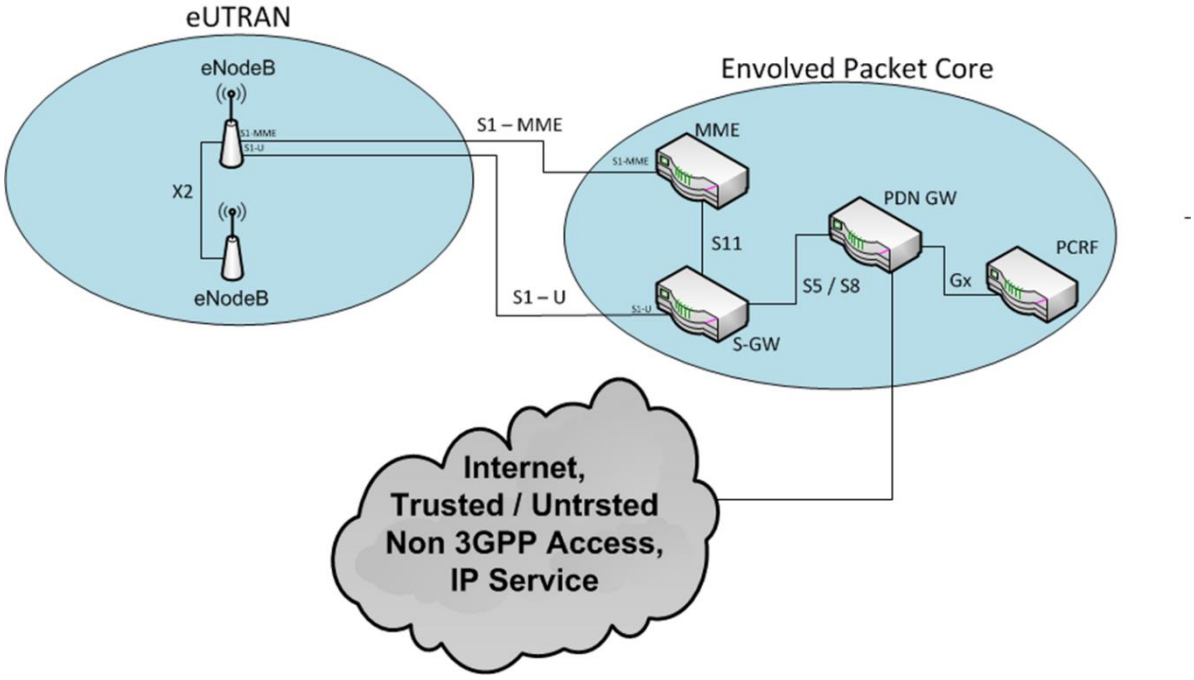


Ilustración 3. Arquitectura de red en LTE [3]

eNodeB – evolución del NodoB, *del inglés evolved NodeB:* Como su nombre así lo dice, es la evolución del elemento NodoB. En comparación con la UTRAN, aquí no está separado el NodoB de la RNC, esta arquitectura cumple ambas funciones de comunicarse directamente con el terminal móvil, así como de tener conexión hacia el CN [2]. Las RNC deben contar con al menos una BTS y los eNodoB deben de ser de la misma marca que la del RNC. Este elemento utiliza las siguientes interfaces para comunicarse con los demás elementos:

- LTE-Uu: para la interconexión entre la eNodeB y el terminal móvil.
- X2: para la interconexión entre los eNodeB.
- S1-MME: para la interconexión entre el eNodeB y el MME.
- S1-U: para la interconexión entre el eNodeB y el S-GW.

EPC – Núcleo de paquetes evolucionado, del inglés Evolved Packet Core: está compuesto por los siguientes elementos:

1) MME: Su función va desde el control del dispositivo móvil realizando la identificación del usuario

en combinación con el HSS hasta la elección del elemento SGW que va a gestionar la comunicación. Este elemento utiliza las siguientes interfaces para comunicarse con los demás elementos:

2) S1-MME: para la interconexión entre el eNodeB y el MME. Referente a la información del control plane.
3) SGs: Para la interconexión entre el MSS y el MME.
4) S11: para la interconexión entre S-GW y MME.
5) S-GW: Elemento que recibe las comunicaciones de datos de los eNodeB. Aísla toda gestión para que no llegue al elemento PGW. Este elemento utiliza las siguientes interfaces para comunicarse con los demás elementos:
    a) S1-U: para la interconexión entre el eNodeB y el S-GW. Referente a la información del plano de usuarios
6) PDN-GW- Puerta de enlace de red de datos en paquetes, *del inglés Packet Data Network Gateway*: Es el elemento que asigna las direcciones IP que utiliza cada usuario por lo que, cara a la red, es como si los datos partieran de él [2]. Este elemento utiliza las siguientes interfaces para comunicarse con los demás elementos:
    a) S5/S8: S5 – información del plano de usuarios entre S-GW y PGW en HPLMN (Inicio Red móvil terrestre pública, *del inglés Home Public Land Mobile Network*).
    b) S8 - información del plano de usuarios entre S-GW y PGW en VPLMN (Red móvil terrestre pública visitada, del inglés Visited Public Land Mobile Network).
7) PCRF – Función de Control de Pólizas y Reglas de Cobro, *del inglés Policy Control and Charging Rules Function*: define las reglas para el cobro y el control de políticas. Es decir, define las acciones y reglas en caso de problemas de incompatibilidad entre la QoS (Calidad del servicio, *del inglés Quality of Service*) que consta en el perfil del usuario y los servicios que se le ofrecen [2].
8) HSS - Servicio al Abonado a Domicilio, *del inglés Home Subscriber Server:* Es la evolución del elemento HLR. Contiene la información del suscriptor como el QoS o perfil de itinerancia. También guarda la información sobre usuario local en MME (es decir, la MME a la cual el UE está conectado) [2].

II.5 Red de registro Single SDB

La red Digitel cuenta con un elemento llamado Single SDB (Base de datos única del subscriptor, *del inglés Single Subscriber Data Base*). Este elemento es una versión ofrecida por Huawei en la cual integra en un mismo elemento físico un HLR y un HSS bajo una base de datos de los

suscriptores unificada. Según la arquitectura de la Corporación, se tiene para los demás elementos de la red un Single SDB (Un HLR y Un HSS) [3]. La solución Huawei para HSS/HLR es el Single SDB.

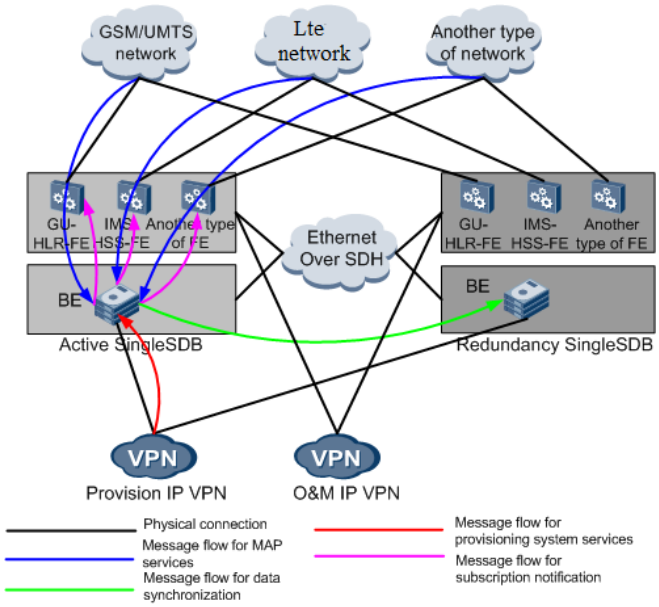


Ilustración 4. redundancia geográfica 2 x (FE + BE) [3]

FE – Parte Delantera, del inglés Front End, unidad lógica que se conecta a la red de señalización y procesa la señalización IP y TDM. El FE no almacena datos de abonado. Obtiene datos del BE.

BE – Parte Trasera, *del inglés Back End*, Unidad lógica que almacena los datos del abonado. Implementa las funciones tales como agregar, eliminar, actualizar y consultar datos basados en los requisitos de procesamiento de servicios de la FE.

Bajo esta información, solo el FE y BE activos proveen servicios. El FE accede al BE local preferiblemente y en condiciones normales solo el BE activo provee servicio.

La Single SDB se conecta al sistema de aprovisionamiento a través de la red IP. Por lo general, los operadores proporcionan una red privada virtual (VPN) dedicada para que el sistema de aprovisionamiento se comunique con la Single SDB. Así, la Single SDB queda aislada de otras redes y se mejora la fiabilidad del sistema.

La Single SDB admite el aprovisionamiento de servicios de múltiples tipos de FE (también conocido como despliegue híbrido de FE). Se puede desplegar un sistema de aprovisionamiento para el aprovisionamiento de servicios de todos los tipos de FE, o se despliega un sistema de aprovisionamiento para el aprovisionamiento de servicios de cada tipo de FE. Los formatos de los comandos que se ejecutan en el sistema de provisión en el despliegue híbrido de FE son los mismos que los formatos de los que se ejecutan en el sistema de provisión en el despliegue autónomo de FE. La ilustración 4 muestra la red del sistema de provisión que soporta la provisión de servicios de todos los tipos de FE.

Actualmente la Corporación Digitel efectúa los cambios y listados de los perfiles de sus abonados a través de la herramienta "Web LMT" (LMT, Terminal de Mantenimiento Local, *del inglés Local Maintenance Terminal*).

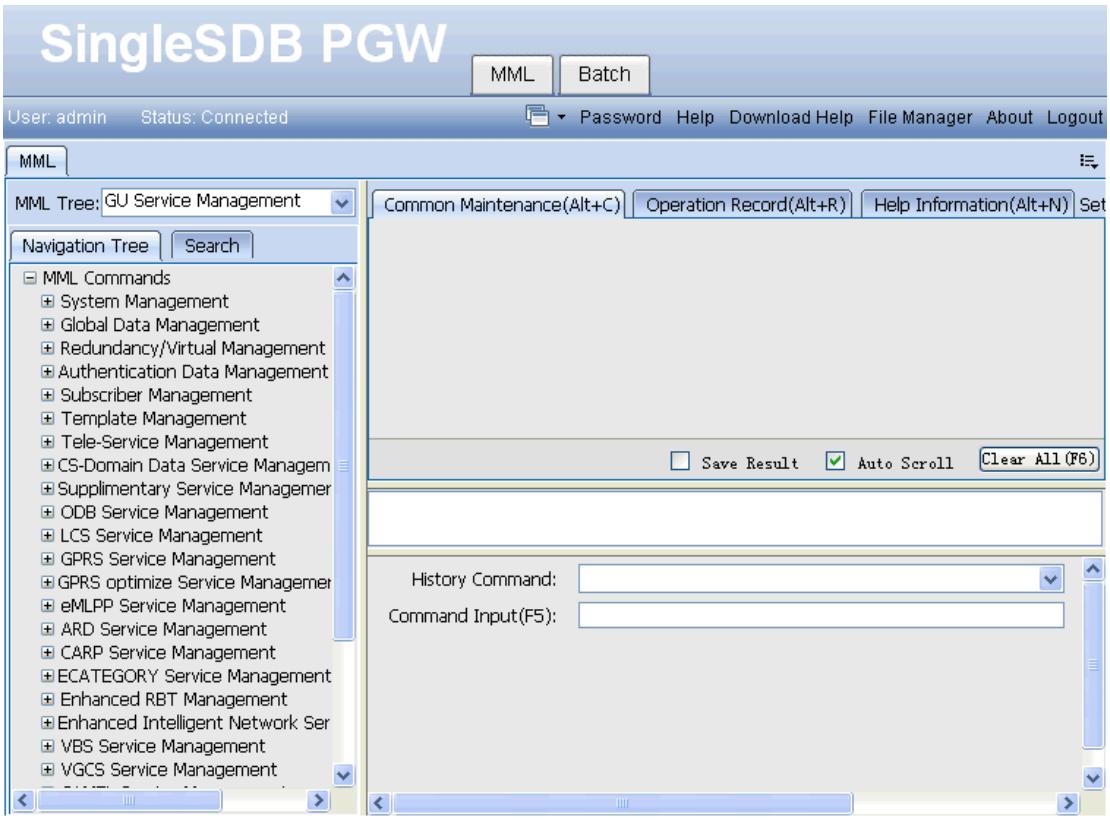


Ilustración 5. Web LMT.

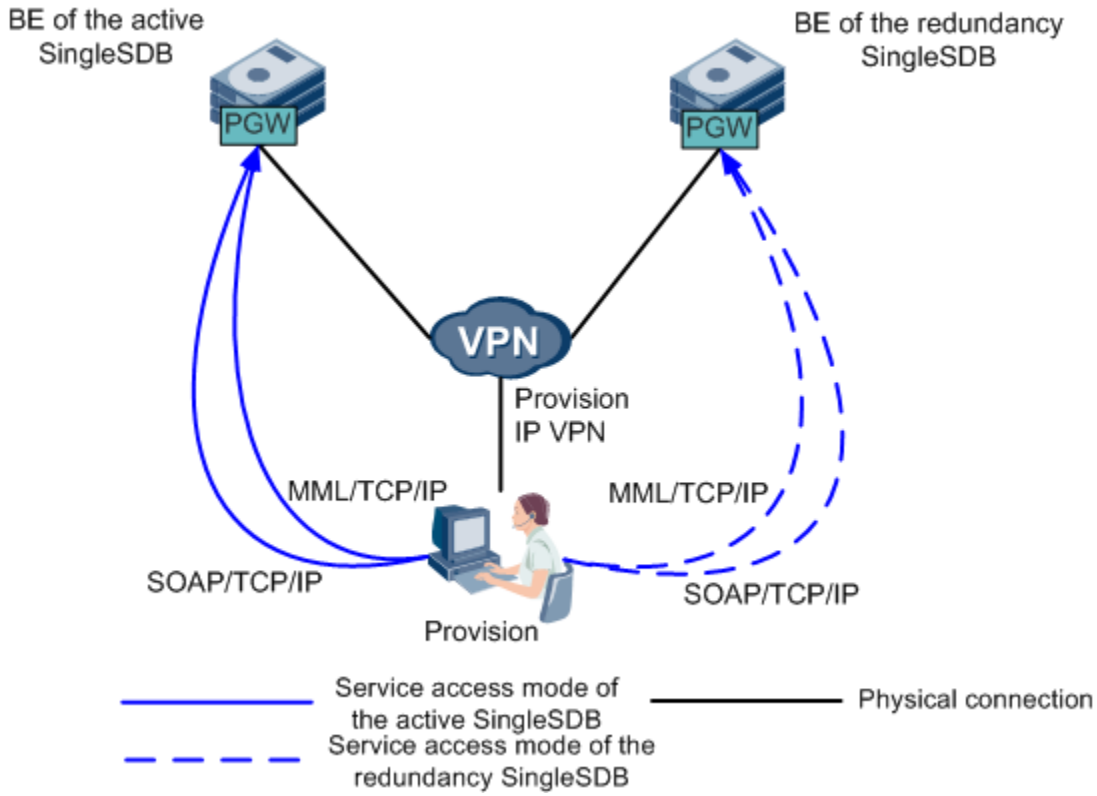


Ilustración 5. Red del Sistema de aprovisionamiento. [3]

El PGW proporciona las mismas direcciones IP y diferentes puertos IP para diferentes tipos de FE. Cada puerto IP soporta diferentes formatos (como el formato MML o SOAP) de comandos enviados desde el sistema de provisión.

Envío de comandos de provisión:

- El sistema de provisión se comunica con la BE de la Single SDB activa a través de TCP/IP.

Sincronización de datos

Después de que la BE de la Single SDB activa ejecute los comandos enviados desde el sistema de aprovisionamiento, los datos relacionados se sincronizan desde la BE de la Single SDB activa a la BE de la Single SDB de redundancia. [3]

II.6 Redes

Una red es una colección de dispositivos llamados nodos, los cuales utilizan protocolos comunes de red para así poder compartir recursos e interactuar entre sí a través de un medio de red. [4]

II.6.1 Tipos de redes:

- PAN - Red de Área Personal, *del inglés Personal Area Network*, permiten a los dispositivos comunicarse dentro del rango de una persona.
- LAN - Red de Área Local, *del inglés Local Area Networks*, son redes privadas que operan en un rango pequeño de operación como una casa, oficina o fábrica. Ampliamente utilizada para conectar computadoras personales y electrodomésticos con el fin de compartir diferentes recursos e intercambiar información. Si se aplica a una empresa, esta es conocida como red empresarial.
- MAN - Red de Área Metropolitana, *del inglés Metropolitan Area Network*, es una red que interconecta usuarios con recursos de computadora en una localización geográfica o una región mayor a la cubierta por una red LAN, pero menor al área cubierta por una red WAN.
- WAN - Red de Área Amplia, *del inglés Wide Area Network*, abarca una extensa área geográfica, por lo general un país o continente.
- VLAN - Red de Área Local Virtual, *del inglés Virtual Local Area Network*, es un método de creación de redes lógicas independientes dentro de una misma red física.

II.7 Protocolos de red

Conjunto de normas estandarizadas que detallan la metodología a emplear para transmitir paquetes de datos entre cualquier cantidad de dispositivos, haciendo de convención que controla y autoriza la conexión, comunicación y transferencia de información. [5]

II.7.1 Protocolos de transporte

Son las reglas que definen cómo se establece la comunicación entre los ordenadores. A decir verdad, algunos protocolos son un conjunto de ellos, donde cada uno se especializa en una tarea definida. Dentro de la capa 4 del modelo OSI se puede apreciar que los protocolos de trasporte son: UDP y TCP. En la tabla 1 se puede apreciar las diferencias entre ellos.

TCP - Protocolo de Control de la Transmisión, *del inglés Transmission Control Protocol:* Es un protocolo de red confiable que está orientado a la conexión, por lo que cada paquete enviado a través de

este protocolo se verifica su correcta entrega, de lo contrario el emisor vuelve a enviar el paquete de datos. Este protocolo también maneja el control de flujo para asegurar que un emisor rápido no pueda inundar a un receptor lento con más mensajes de los que pueda manejar [5].

UDP - Protocolo de Datagrama de Usuario, *del inglés User Datagram Protocol:* Es un protocolo de red no confiable, ya que no está orientado a la conexión, por lo que el emisor no es consciente de si el paquete se entregó de manera adecuada ya que no hay verificación de entrega como lo hace TCP. También se utiliza mucho en las consultas de petición-respuesta de una sola ocasión del tipo cliente-servidor, y en las aplicaciones en las que es más importante una entrega oportuna que una entrega precisa, como en la transmisión de voz o video [5].

	TCP	UDP
Sentido	Establece conexión entre los ordenadores antes de transmitir datos	Envía los datos directamente al ordenador de destino sin verificar si el sistema está listo para recibir o no
Se expande a	Protocolo de Control de Transmisión	Protocolo de datagramas de usuario
Tipo	Orientado a la conexión	No orientado a la conexión
Velocidad	Lento	Rápido
Confiabilidad	Altamente fiable	No fidedigno
Tamaño del encabezado	20 bytes	8 bytes
Reconocimiento	Requiere reconocimiento de datos y tiene la capacidad de volver a transmitir si el usuario lo solicita	No requiere reconocimiento de datos

1 Tabla comparativa TCP vs. UDP

II.7.2 NTP

"El protocolo de sincronización determina el desfase horario de un reloj del cliente con respecto a uno o varios relojes del servidor. los distintos protocolos de sincronización que se utilizan hoy en día

proporcionan diferentes medios para hacerlo, pero todos siguen el mismo modelo general. el cliente envía una solicitud al servidor y éste responde con su hora actual. para obtener la máxima precisión, el cliente necesita medir el retraso de propagación del cliente del servidor para determinar el verdadero desfase horario con respecto al servidor. Como no es posible determinar los retrasos en un sentido, a menos que se conozca el desfase real, NTP mide el retraso total de ida y vuelta y asume que los tiempos de propagación son estadísticamente iguales en cada dirección. En general, esta es una aproximación útil; sin embargo, en la Internet de hoy, las rutas de la red y los retrasos asociados pueden diferir significativamente, causando errores de hasta la mitad de la diferencia de retardo de la ruta". [6, pp. 3,4]

II.7.3 Almacenamiento sobre redes IP: ISCSI

Una de las actividades principales de un centro de computadoras o cualquier otro que soporte servidores, es proveer una adecuada capacidad de discos de almacenamiento. Los canales de fibra óptica suelen ser utilizados para este propósito, sin embargo, ISCSI (Internet SCSI) provee una solución de bajo costo al canal de fibra que puede aprovechar los servidores. [7]

ISCSI es un protocolo de red de almacenamiento que simplifica la transferencia de paquetes de datos sobre redes TCP/IP entre servidores y dispositivos de almacenamiento en bloques. El software de destino ISCSI se ejecuta en el servidor de destino y define las unidades lógicas como dispositivos de destino ISCSI. El software iniciador de ISCSI se ejecuta en diferentes servidores y se conecta a los dispositivos de destino para que los dispositivos de almacenamiento estén disponibles en los servidores iniciador.

El servidor de destino ISCSI LIO y los servidores iniciadores ISCSI se comunican enviando paquetes SCSI a el nivel de IP en su LAN. Cuando una aplicación que se ejecuta en el servidor iniciador inicia una búsqueda de un dispositivo de destino ISCSI LIO, el sistema operativo produce los comandos SCSI necesarios. Los comandos SCSI son entonces incrustados en paquetes IP y encriptados según sea necesario por el software iniciador ISCSI. Los paquetes se transfieren a través de la IP interna a la correspondiente estación remota ISCSI, llamada servidor de destino ISCSI LIO, o simplemente el objetivo del ISCSI.

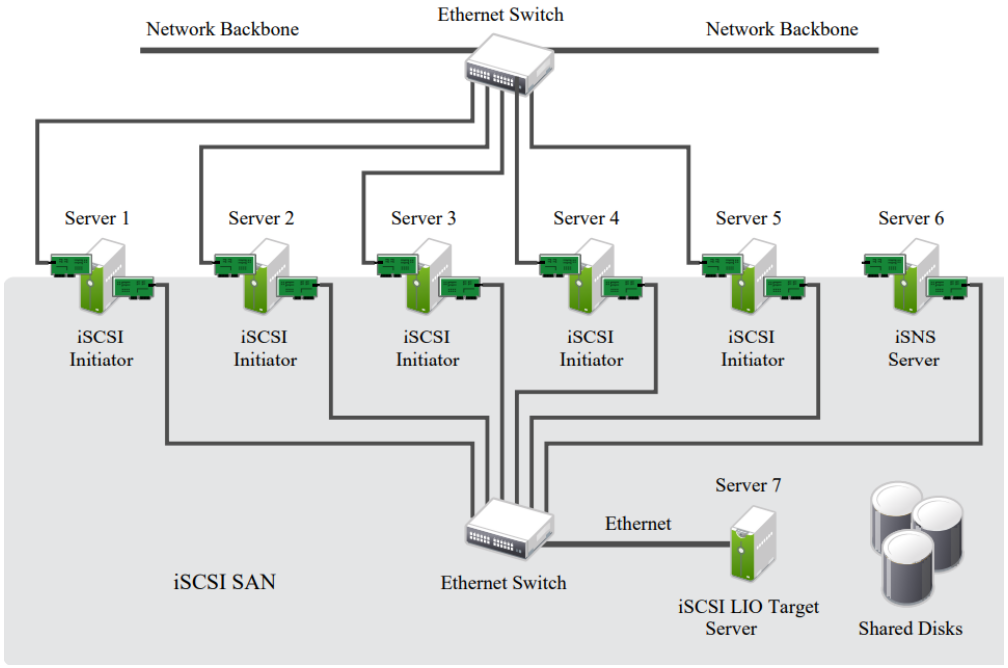


Ilustración 6. ISCSI SAN con un servidor ISNS [7]

II.8 Protocolos en el nivel de aplicación

Es la última capa del modelo OSI y proporciona servicios a los usuarios tales como:

II.8.1 SSH – intérprete de órdenes seguro

La Carcasa Segura, *del inglés Secure Shell;* es un enfoque popular, poderoso y basado en software para la seguridad de la red. Siempre que un ordenador envía datos a la red, SSH los cifra automáticamente. Cuando los datos llegan a su destinatario, SSH los desencripta automáticamente (descifra). El resultado es un cifrado transparente: los usuarios pueden trabajar normalmente, sin saber que sus comunicaciones están cifradas de forma segura en la red. Además, SSH utiliza algoritmos de encriptación modernos y seguros y es lo suficientemente efectivo como para ser encontrado en aplicaciones de misión crítica en las principales corporaciones. [8, p. 20]

Algunas características son:

- Después de la conexión inicial, el cliente puede verificar que se está conectando al mismo servidor al que se conectó anteriormente.
- El cliente transmite su información de autenticación al servidor usando una encriptación robusta de 128 bits.

- Todos los datos enviados y recibidos durante la sesión se transfieren por medio de encriptación de 128 bits, lo cual los hacen extremamente difícil de descifrar y leer.
- El cliente tiene la posibilidad de reenviar aplicaciones X11 desde el servidor. Esta técnica, llamada reenvío por X11, proporciona un medio seguro para usar aplicaciones gráficas sobre una red.
- SSH utiliza el puerto 22 de TCP, sin embargo, se puede modificar si es deseado.

II.8.2 SFTP – Protocolo de transferencia segura de archivos, Del inglés Secure File Transfer Protocol

El Protocolo de Transferencia Segura de Archivos, del inglés Secure File Transfer Protocol; es una herramienta separada de transferencia de archivos en capas sobre SSH. Fue desarrollado por SSH Communications Security y originalmente sólo estaba disponible en SSH2, pero desde entonces han aparecido otras implementaciones. [8, p. 60]

Algunas características son:

- Permite la realización de diferentes operaciones sobre archivos remotos.
- Se aplica con más frecuencia en plataformas Unix, aunque existen servidores SFTP en la mayoría de las plataformas.
- Está diseñado para ser un protocolo independiente.
- No es aún un estándar de Internet.
- La versión más utilizada es la versión 3, ejecutada por el popular servidor OpenSSH de SFTP.
- En su versión 4, redujo sus vínculos con la plataforma Unix, por lo que muchos sistemas operativos Windows basan sus implementaciones en servidores SFTP.
- SFTP utiliza el puerto 22 de TCP.
- La seguridad en la transferencia no la provee directamente el protocolo SFTP, sino SSH o el protocolo que sea utilizado en su caso para este cometido.
- Para subir archivos, los archivos transferidos pueden estar asociados con sus atributos básicos, como el de tiempo, esta última es una ventaja sobre el protocolo FTP común, ya que no dispone de ningún crédito para incluir archivos en la fecha original.
- Los programas de SFTP ofrecen para los clientes que los utilizan una interfaz interactiva similar a la de los tradicionales programas de FTP.

II.9 Servidor

"Los servidores son equipos informáticos que brindan un servicio en la red. Dan información a otros servidores y a los usuarios". [9, p. 23] Son equipos de mayores prestaciones y dimensiones que una computadora personal. Actualmente existen diversos tipos de servidores, pero solo se profundizará en el utilizado en este proyecto.

Uno de los componentes de un servidor es el semiconductor de óxido metálico complementario (CMOS) es una pequeña cantidad de memoria en la placa base de un ordenador que almacena la configuración del sistema básico de entrada/salida (BIOS). Cuando el servidor esta apagado, este componente mantiene las configuraciones del BIOS gracias a una batería BR 2032 de 3V.

II.9.1 Servidor espejo

Un servidor espejo, como lo indica su nombre, es una réplica exacta del servidor principal, siendo capaz de sustituirlo en caso de que este tenga alguna falla u error durante su funcionamiento. Esto puede ser uno o más archivos, una base de datos, un sitio web o un servidor entero. El servidor espejo es perfectamente capaz de replicar los servicios del servidor principal, ya que poseen la misma información y configuración, lo que garantiza una alta disponibilidad del servicio y por ende mejora la calidad del servicio (QoS) a ofrecer. Ésta es su principal funcionalidad, pero también puede aprovecharse para distribuir mejor las cargas de trabajo de cada una de las máquinas, ya que las dos estarán siempre accesibles y sincronizadas. Además, los dos servidores no tienen por qué estar en la misma ubicación física, por lo que, ante problemas de fallos energéticos locales, nos aseguramos que al menos uno de los dos estará disponible para nuestros clientes o empleados. [10]

II.9.2 Sun Netra X4270

Este diseño de servidor se enfoca en misiones críticas de computación y eco responsabilidad. Estos potentes y ampliables sistemas 2U de grado portador utilizan Intel Xeon de alto rendimiento y ofrecen una avanzada tecnología de computación, memoria, almacenamiento y densidad de I/O. Al mismo tiempo desempeñan una gran eficiencia energética sobre plataformas que concuerdan en las certificaciones para Telecordia NEBS (Norma de Construcción de Equipos de Red, del inglés Network Equipment Building Standard) nivel 3. Ideales para virtualización e iniciativas de consolidación. [11]

Entre las aplicaciones más comunes se encuentran:

- Controlador de Media Gateway.
- Sistemas de mantenimiento y operación para redes de telecomunicaciones.

- Gateway de señalización.
- Redes inteligentes.
- MMS (servicios de mensajería multimedia) / SMS (servicio de mensajería corta).
- Defensa/militar/inteligencia.
- Servidor de aplicaciones.
- Servidor web.
- Servidor proxy.
- Home/visitor location register (HLR/VLR).
- Controladores de estaciones base (BSC).
- Redes de distribución de contenidos.
- Servidor DNS.
- Cortafuegos para redes virtuales privadas / seguridad IP (VPN/IPSEC)
- Sistemas de seguridad.

No.	Description	Additional Information
1	User alarm status LEDs	Top to bottom – Critical LED, Major LED, Minor LED, User LED
2	System status LEDs	Left to right – Locator LED button, Service Required LED, System Activity LED, Power button
3	Removable media	Only in two drive configurations

Ilustración 7. Servidor Sun Netra X4270 [11]

II.9.3 SUSE Linux Servidor Empresarial extensión 15.1

SUSE Linux es una de las más conocidas distribuciones Linux existentes a nivel mundial, se basó en sus orígenes en Slackware. Su nombre “SUSE” es el acrónimo, en alemán “Software und Systementwicklung” (Desarrollo de Sistemas y de Software), el cual formaba parte del nombre original de la compañía y que se podría traducir como “desarrollo de software y sistemas”. [12]

SUSE incluye un programa único de instalación y administración llamado YaST2 que permite realizar actualizaciones, configurar la red y el cortafuego, administrar a los usuarios, y muchas más opciones todas ellas integradas en una sola Interfaz amigable. Además, incluye varios escritorios, entre ellos los más conocidos que son KDE y Gnome, siendo el primero el escritorio por omisión. La distribución incorpora las herramientas necesarias para redistribuir el espacio del Disco duro permitiendo así la coexistencia con otros sistemas operativos existentes en el mismo.

Usa sistemas de paquetes RPM (RPM package manager) originalmente desarrollados por Red Hat aunque no guarda relación con esta distribución.

Distribuciones Linux basadas en SUSE Linux:

- Novell Linux Desktop.
- Java Desktop System.
- SUSE Linux Enterprise Server Edition. (Edición tentativa).

II.9.4 RAID

Matriz redundante de discos independientes, *del inglés Redundant Array of Independent Disk* es un sistema de seguridad y de integridad en sistemas informáticos (sobre todo utilizado en servidores), que permite discos duros espejo.

"La idea de RAID se basa en la combinación de múltiples unidades de disco pequeñas y poco costosas que se agrupan en una formación para lograr objetivos de mejor rendimiento o redundancia que no se pueden lograr con una única unidad grande y costosa. Esta formación de discos el ordenador las considerará como si fueran una única de disco lógica". [13]

Existen diferentes tipos de configuraciones de RAID para diversas aplicaciones prácticas, pero para el desarrollo de este tomo solo utilizaremos RAID 1, ya que es la configuración a usar para crear redundancia de almacenamiento y por lo tanto corregir cualquier falla que se pueda presentar en uno de los discos. Por ende, se dejarán por fuera las demás configuraciones a excepción de una breve imagen que resume los diversos posibles arreglos de discos.

- RAID 1: Es la configuración más utilizada de RAID, ya que replica los datos de un disco en otro que haría de respaldo en caso de que un disco falle, el respaldo sigue funcionando. A esta técnica se le conoce por su simplicidad y su elevada tasa de transferencia de datos de lectura, aunque normalmente actúan de manera independiente y generan altos niveles de transferencia de datos de entrada y salida I/O.

Cuando entramos en la zona de detección de discos en la BIOS, podemos seleccionar AUTO en la columna del tipo de disco (TYPE) para que cada vez que el ordenador arranque se coloquen los valores automáticamente. Los discos duros SCSI no se registran en la BIOS del sistema. En cambio, la tarjeta adaptadora SCSI incluye su propio BIOS (llamado web BIOS), que regulará todas las actividades, con independencia del microprocesador. Un ordenador que tenga tarjeta adaptadora SCSI mostrará información acerca del adaptador en el proceso de arranque y también la posibilidad de acceder a la BIOS del adaptador mediante una combinación de teclas, como, por ejemplo, [Ctrl + H].

TIPO DE RAID	DETALLES
RAID 0	Se graban los datos distribuidos, pero sin tolerancia a fallos.
RAID 1	Se graban los datos en espejo; si un disco falla, el otro sigue funcionando.
RAID 2	Utiliza un algoritmo complicado que demanda muchos cálculos a la CPU. Es lento y se requieren discos especiales. Permite acceso en paralelo.
RAID 3	Usa un disco de control de paridad, de esta forma permite el acceso en paralelo, pero todos los discos deben funcionar al unísono.
RAID 4	Es parecido al RAID 3, pero es posible acceder a los sectores de forma individual. No es necesario leer de todos los discos al mismo tiempo.
RAID 5	Se graba en forma distribuida con datos de paridad para controlar los datos; si cualquier disco se rompe, el sistema sigue funcionando.
RAID 6	Funciona de manera parecida al RAID 5, pero permite que se rompan dos discos. Debemos tener en cuenta que es muy poco utilizado debido a su elevado costo.

Ilustración 8. Niveles de RAID con sus respectivos detalles. [9, p. 39]

II.9.5 Web BIOS

La utilidad de configuración de Web BIOS (CU) proporciona una utilidad basada en web para configurar y gestionar volúmenes RAID. La utilidad configura los arreglos de discos y unidades lógicas. Su funcionamiento es independiente del sistema operativo porque la utilidad reside en la BIOS Mega RAID. [14]

La Web BIOS CU realiza las siguientes acciones:

- Muestra las propiedades del adaptador.
- Escanea los dispositivos.
- Crea arreglos físicos.
- Define las unidades lógicas.
- Muestra las propiedades lógicas de las unidades.
- Inicializa las unidades lógicas.
- Comprueba la consistencia de los datos.

- Muestra las propiedades físicas de los dispositivos.

II.9.6 Clúster de alta disponibilidad

El objetivo de un clúster de alta disponibilidad es asegurar que los recursos críticos alcancen la máxima disponibilidad posible. Este objetivo se logra instalando software de clúster en múltiples servidores (Figura 6). Este software supervisa la disponibilidad de los nodos del clúster, y monitorea la disponibilidad de los servicios que son gestionados por el clúster (en este caso, estos servicios se denominan recursos). Si un servidor presenta fallas y detiene su servicio, el clúster HA (alta disponibilidad, *del inglés high available*) se dará de cuenta y se asegurará de que el recurso sea reiniciado en algún otro lugar del clúster, para que así se pueda volver a usar el recurso después de una mínima interrupción. [15]

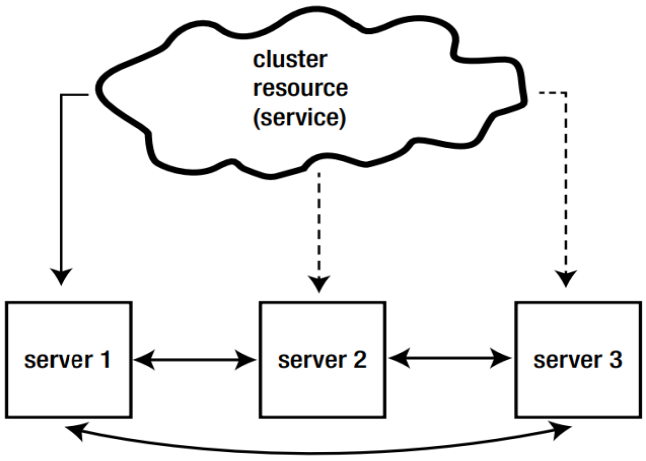


Ilustración 8. Heartbeat [12]

II.9.7 Extensión de Alta Disponibilidad de Suse Linux Extensión 15.1

Esta extensión es una suite integrada por tecnologías de clustering de código abierto (open source), el cual permite implementar clústeres Linux físicos y virtuales de alta disponibilidad, y eliminar así puntos únicos de falla. Asegurando alta disponibilidad y manejabilidad de recursos críticos de redes incluyendo data, aplicaciones y servicios, por lo tanto, le ayuda a mantener la continuidad del servicio protegiendo la integridad de la data y reduce los tiempos fuera de servicio.

Esta extensión está disponible para SUSE Linux Enterprise server 15 sp1. [16]

II.9.8 Hawk

Es una interfaz web amigable con el usuario que permite monitorear y administrar los clústeres de alta disponibilidad desde ordenadores sin importar su sistema operativo. Se puede acceder a Hawk desde cualquier navegador web que este dentro o fuera de los clústeres. [16]

II.9.9 Corosync

El motor de clúster de Corosync es un sistema de comunicación de grupo con características adicionales para implementar una alta disponibilidad en las aplicaciones. Corosync es utilizado como un framework de alta disponibilidad por proyectos como Apache, Qpid y Pacemaker. [16]

Esta herramienta posee las siguientes características:

- Sincronía virtual: un modelo de comunicación de grupo de procesos cerrados con sincronía virtual que garantiza la creación de máquinas de estado replicadas.
- Disponibilidad: un sencillo gestor de disponibilidad que reinicia el proceso de la aplicación cuando ha fallado.
- Información: una base de datos de configuración y estadísticas en memoria que proporciona la capacidad de establecer, recuperar y recibir notificaciones de cambio de información.
- Quorum: un sistema de quorum que notifica a las aplicaciones cuando se alcanza o se pierde el quorum.

II.9.10 STONITH

Es un acrónimo en inglés para Dispárale al Otro Nodo en la Cabeza, del inglés *shoot the other node in the head*. Su función es proteger los datos de que se corrompan por un nodo rebelde.

El hecho de que uno de los nodos no responda, no significa que este no esté accediendo a sus datos y la única forma de garantizar la integridad de los mismos es cercando al nodo rebelde a través de stonith para asegurarse de que el nodo este fuera de línea antes de permitir el acceso a los datos desde otro nodo.

En caso de un grupo de servicios no responda adecuadamente, stonith se encargará de cercar el nodo en el que se presenta la falla y se velará por que se inicie el mismo grupo de servicios en otro nodo que esté disponible. [17]

II.9.11 SBD

Dispositivo de bloques STONITH – *del inglés STONITH Block Devices*, "provee un mecanismo de cercado de nodos para clústeres basado en Pacemaker a través del intercambio de mensajes de bloques compartidos de almacenamiento (SAN, ISCSI, FCoE, etc.). Esto aísla el mecanismo de cercado de los

cambios en la versión de firmware o dependencias en controladores específicos de firmware. SBD necesita del iempo del perro guardián – *del inglés Watchdog Time*, en cada nodo para asegurar que los nodos rebeldes sean detenidos realmente". [16]

La prioridad del arreglo de clústeres de alta disponibilidad es proteger la integridad de los datos, esto se logra al prevenir el acceso no coordinado al almacenamiento de datos, de lo cual se encarga el arreglo de clústeres a través de distintos mecanismos. Sin embargo, mal funcionamiento de equipos físicos o la partición de la red podrían crear escenarios en donde varios DC's (Coordinador Designado, *del inglés Designated Coordinator*) son elegidos en un clúster. Esto se conoce como Escenario de Cerebros Divididos, *del inglés Split Brain Scenario*, el cual podría ocasionar la corrupción de los datos.

El mecanismo utilizado para prevenir que este escenario se desarrolle es el cercado de los nodos a través de STONITH. SBD es una alternativa de cercado de nodos cuando no se cuenta con fuentes externas de apagado en caso de presentarse un escenario de cerebros divididos.

II.9.11.1Componentes y mecanismos SBD

- Partición SBD: En un ambiente donde todos los nodos tienen acceso a una memoria compartida, una pequeña partición es formateada para usar con SBD. El tamaño de la partición dependerá del tamaño del bloque utilizado en el disco (1 MB para discos ISCSI con bloques de 512 byte o 4 MB para discos DASD con bloques de 4 kB byte). El proceso de inicialización crea una capa de mensajes en el dispositivo con capacidad de 255 nodos.
- Demonio SBD: después de que el respectivo demonio SBD es configurado, este es activado en cada nodo antes de que el resto de los equipos del clúster sean activados y, es desactivado después de que todos los componentes restantes del clúster sean apagados, asegurando así que los recursos del clúster no sean activados sin la supervisión de SBD.
- Mensajes: El demonio asigna automáticamente una de las ranuras de mensajes para sí mismo y la monitoriza constantemente para los mensajes dirigidos a sí mismo. Al recibir un mensaje, el demonio cumple inmediatamente con la petición, como iniciar un ciclo de apagado o de reinicio para efectuar el cercado. Además, el demonio monitorea constantemente la conectividad al dispositivo de almacenamiento, y se termina a sí mismo en caso de que la partición se vuelva inalcanzable. Esto garantiza que no esté desconectado de los mensajes de cercado.
- Watchdog: Siempre que se utilice SBD, es crucial que el watchdog funcione adecuadamente. Los sistemas modernos soportan este hardware que necesita ser alimentado por un componente de software (en este caso el demonio SBD), el cual regularmente escribe un pulso de servicio al watchdog y, si este deja de ser alimentado por el demonio, el hardware va a obligar al sistema

a reiniciarse. Esto protege contra fallas del proceso SBD, como morir, o quedarse atascado en un error de I/O (input/output).

II.9.12 Heartbeat

Es una red privada que solo es compartida por los nodos del clúster y no es accesible fuera del mismo. Este protocolo es utilizado por los nodos del clúster para monitorear el estado de cada nodo y comunicarse entre sí a través de mensajes de control necesarios para mantener el funcionamiento del mismo.

Este protocolo utiliza la metodología FIFO *(*Primero en Entrar es el Primero en Salir – *del inglés First In First Out),* de las señales enviadas a través de la red. Asegurándose de que todos los mensajes han sido recibidos, el sistema asegura que los eventos pueden ser ordenados correctamente, estos mensajes son vistos como mensajes de control que ayudan a determinar que la red no incluye mensajes atrasados. [16]

II.10 Perl

Perl (Extracción práctica y lenguaje de informes, del inglés Practical Extraction and Report Language) es un lenguaje de programación desarrollado a finales de los años 80 por Larry Wall a partir otras herramientas de UNIX como son: ed, grep, awk, c-shell, para la administración de tareas propias de sistemas UNIX. [18]

Es un lenguaje de script de tipo BCPL (como TCL o PHP), muy semejante al AWK (de hecho está basado en él), de tipo estructurado con trazas de orientación a objetos (no completamente soportado de forma directa), que permite el desarrollo rápido de aplicaciones y herramientas especialmente orientadas al tratamiento de textos y archivos, aunque actualmente también se utiliza incluso para entornos gráficos, en combinación con sistemas como Perl/TK o GTK.

Características:

Estructuralmente, Perl está basado en un estilo de bloques como los del C o AWK, y fue ampliamente adoptado por su destreza en el procesado de texto y no tener ninguna de las limitaciones de los otros lenguajes de script.

- No establece ninguna filosofía de programación concreta. No se puede decir que sea orientado a objetos, modular o estructurado, aunque soporta directamente todos estos paradigmas y su punto fuerte son las labores de procesamiento de textos y archivos.
- No es ni un compilador ni un intérprete, está en un punto intermedio, cuando mandamos a ejecutar un programa en Perl, se compila el código fuente a un código intermedio en memoria que se optimiza como si se fuera a elaborar un programa ejecutable, pero es ejecutado por un motor, como si se tratase de un intérprete.
- Lenguaje de programación basado en scripts a casi cualquier plataforma.
- Lenguaje optimizado para el escaneo de texto arbitrario de ficheros. Es también un buen lenguaje para tareas de administración de sistemas. Es un lenguaje con intención de ser práctico en lugar de bonito.
- Básicamente, es un lenguaje que se ha intentado por parte de su creador que sea lo más natural posible, lo que conlleva que en ocasiones nos encontremos estructuras poco habituales en un lenguaje de este tipo.

Capítulo III

Metodología

Este trabajo especial de grado se define como un proyecto especial, debido a la necesidad que tiene la corporación Digitel de implementar un servicio capaz de optimizar el tiempo con el que se puede generar consultas o modificaciones a la base de datos que alberga los perfiles de todos los abonados de la red (Single SDB), para ello se requirió modificar unos scripts previamente existentes en lenguaje Perl, permitiendo ejecutarlos e introducir el comando deseado a través de la terminal de un servidor con arreglo en espejo, que a su vez cada uno de ellos cuenta con un arreglo de discos RAID 1 y software de alta disponibilidad en ambiente Linux. El arreglo de los equipos físicos se realizó con la intención de garantizar la calidad del servicio acorde a una empresa de Telecomunicaciones en caso de fallas.

III.1 Fase I: Investigar antecedentes bibliográficos y tecnologías sobre el tema y la red Digitel.

Se realizaron investigaciones en fuentes académicas comprobadas (libros, publicaciones, manuales técnicos, documentación detallada y confidencial de la empresa, etc.) referentes a los temas claves para el desarrollo de este trabajo, destacando la alta disponibilidad y replicación de información, en conjunto a la red de la corporación, características de los diferentes equipos a manipular como el Single SDB, antecedentes y beneficios, al igual que los mecanismos de QoS para el servicio a través del servidor espejo, arreglo de discos y la alta disponibilidad (HA).

También, se revisaron documentos técnicos de los equipos disponibles en la Corporación Digitel en esta fase para el desarrollo del proyecto.

- Investigar acerca de los equipos dentro de la red con los que se requiere establecer una conexión para lograr la ejecución de los scripts.
- Investigar acerca de los servidores, servidor espejo, software de alta disponibilidad (HA), replicación de datos, arreglo de discos y protocolos de red.
- Investigar acerca del lenguaje de programación Perl para poder manipular, modificar e implementar los scripts necesarios para el desarrollo del proyecto.

III.2 Fase II Investigar la operación actual de la red DIGITEL

Se profundizó a detalle el Single SDB, obteniendo detalles clave como dirección IP y puertos a través de los cuales se efectúa la comunicación. Estos se utilizan en los scripts para obtener acceso al equipo de forma remota, aplicando conocimientos de telemática y de programación para que al ejecutar los scripts se ingrese de manera automática a Single SDB y se ejecute el comando deseado previamente introducido por la terminal del servidor implementado.

III.3 Fase III: Determinar cuáles son las consultas en la HLR y HSS de mayor importancia para realizarlas a través de un programa o código desarrollado a través de un lenguaje de programación.

En esta fase se determinó que todas las posibles consultas o modificaciones son indispensables para la rutina del departamento de Operación y mantenimiento (O&M), motivo por el cual se profundizó el conocimiento acerca del lenguaje de programación Perl para ofrecer una solución que contemple la posible ejecución de cualquier consulta deseada por el usuario a través de comandos introducidos por teclado en la terminal de forma local o remota.

III.4 Fase IV: Diseño del sistema tentativo a implementar.

Una vez obtenidos los conocimientos técnicos necesarios, y habiendo verificado la documentación de los equipos disponibles, se analizó el diseño planteado para la puesta en marcha del proyecto, desarrollando un sistema que permitiría escalabilidad.

En esta fase se realizaron las siguientes actividades:

- Diseño inicial del sistema acorde a lo que se requiere y dispone.
- Análisis de las características de los equipos provistos.
- Softwares requeridos.

III.5 Fase V: Implementación

Una vez definido el sistema a emplear, se puso en práctica los conocimientos adquiridos para el desarrollo de esta fase siguiendo los lineamientos de configuración de los equipos, infraestructura y configuración de los servidores acompañados del software de alta disponibilidad.

Las siguientes actividades fueron asignadas a esta fase:

- Configuración e instalación de los equipos.
- Conexión de los equipos a la red de la Corporación y crear redundancia entre los servidores del proyecto.
- Configuración de la matriz redundante de discos (RAID 1) y del software de alta disponibilidad (HA).
- Configuración de un disco externo a través del protocolo ISCSI que alimente al demonio SBD con el objetivo de cercar los nodos en caso de ocurrir alguna falla, haciendo que el nodo inicie el ciclo de reinicio.

III.6 Fase VI: Pruebas de funcionamiento y operatividad del servidor.

En esta fase se pretende llevar a cabo un conjunto de pruebas con la finalidad de detectar fallas y efectuar las correcciones pertinentes con aras a un correcto funcionamiento en todos los componentes que comprenden este sistema y desarrollar un análisis en base a resultados obtenidos.

Entre las pruebas efectuadas se tienen:

1. Prueba de discos
2. Redundancia de puertos
3. Resolución de Hostname o nombre del hospedador (servidor)
4. Acceso remoto
5. Acceso remoto a través de la IP virtual
6. Reinicio o cercado del nodo a través del software de alta disponibilidad.
7. Ejecución de Scripts con lista de abonados

III.7 Fase VII: Elaboración del tomo del trabajo especial de grado.

En esta fase se utilizó toda la información recopilada a lo largo del desarrollo de las fases y el capítulo II con el propósito de ser utilizada para la elaboración del tomo a entregar.

Capítulo IV

Desarrollo

A continuación, se describe en detalle la forma en que se desarrollaron las diferentes actividades, por fases, las cuales fueron los lineamientos para dar como resultado el cumplimiento en su totalidad de los objetivos planteados anteriormente.

IV.1 Investigar antecedentes bibliográficos y tecnologías sobre el tema y la red Digitel.

Para este Trabajo Especial de Grado, se desarrolló la búsqueda, recopilación y ampliación de conocimientos de todos los aspectos teóricos que sirvieron de soporte para abordar el problema planteado: Implementar un sistema a través del cual se pueda consultar el registro de usuarios dentro de la Corporación Digitel.

La investigación se centra, mayormente, en libros y manuales digitales, los cuales se encuentran debidamente citados. Se efectuó una investigación acerca del Single SDB, obteniendo de este su dirección IP, puerto de acceso y credenciales de acceso. Luego se procedió a investigar acerca de los servidores y posibles configuraciones, pasando por los arreglos de discos hasta llegar al software de alta disponibilidad (HA) para la replicación de datos.

Un aspecto fundamental para la documentación fue el estudio del concepto de registro de usuarios, como establecer un canal de comunicación para acceder y modificar dicho registro a través de un lenguaje de programación que se adapte a la necesidad del proyecto para manipular procesos internos del servidor y por cual medio. Una vez culminada esta parte, se procedió a investigar sobre servidores (S.O., características y alcance) para luego proceder con la replicación de información a través de un servidor espejo utilizando software de alta disponibilidad (HA) y arreglo de discos RAID 1 para crear redundancia en los discos al igual que en los servidores.

Al concluir, toda la recolección de documentos permitió obtener una base teórica sólida que permitió desarrollar el diseño e implementación del proyecto a realizar.

- Investigar acerca del equipo HLR/HSS dentro de la red de la Corporación Digitel para establecer una conexión con el objetivo de la ejecución de los scripts.
- Investigar acerca de los servidores y S.O., servidor espejo, software de alta disponibilidad (HA), replicación de datos, arreglo de discos y protocolos de red.

- Investigar acerca del lenguaje de programación Perl para lograr implementar los scripts necesarios para desarrollar el proyecto.

Al culminar, la totalidad de documentos recolectados derivó en la obtención de una base teórica sólida que permitió desarrollar el diseño e implementación del proyecto a efectuar.

IV.1.1 Análisis sobre los equipos dentro de la red con los que se desea establecer comunicación

Se procedió a solicitar documentación detallada acerca del Single SDB, con el objetivo de entablar una comunicación a través de la cual se busca efectuar consultas o modificaciones al registro de usuarios de la corporación. Para esto se requirió obtener la dirección IP del Single SDB, el puerto a través del cual se entabla la comunicación y las credenciales necesarias para acceder al mismo. Adicionalmente se requirió efectuar un estudio de los comandos internos del PGW para manipular el registro de usuarios acorde a las necesidades del departamento a través de los scripts en lenguaje PERL.

IV.1.2 Análisis de posibles soluciones para replicar datos y alta disponibilidad

En primera instancia se indagó sobre el hardware del equipo, con la intención de escoger el sistema operativo que mejor se adapte a las necesidades del proyecto y que a su vez sea software libre para reducir los costos del mismo.

Entre las características del equipo se encuentran:

Memoria Ram	48 GB
Procesador	2 x Quad-core Intel Xeon Processor 5500 Series (L5518)
Almacenamiento	2x300GB
Puertos de Red	4 x 10/100/1000 Mb
Expansión PCI	Si
# Fuentes de poder	2

Tabla 2, Hardware de los Servidores utilizados.

Acorde a las características del Hardware se buscó un sistema operativo en base a Linux que se adapte a las necesidades del proyecto, entre los se encuentran:

- Suse Linux Enterprise Server.
- Ubuntu.
- Debian.

Adicionalmente se efectuó un análisis enfocado en los diversos softwares de alta disponibilidad (HA) para lograr la replicación de datos, esto con el objetivo de obtener redundancia en el sistema para que en caso de fallar uno de los servidores, el otro puede ser puesto en funcionamiento de forma inmediata y autónoma.

Entre los softwares de alta disponibilidad que es estudiaron se encuentran:

Software	Versión	Compatibilidad
Suse Linux Enterprise High Availability Extension	11.4	X86
	15.1	AMD64/Intel 64
Red Hat High Availability Add-on	8	64-bit x86 or ARM System.

Tabla 3, softwares de alta disponibilidad

A pesar de que Red Hat ofrece un software de alta disponibilidad con una sólida documentación y foros, este tuvo que ser descartado debido a que el precio varía, iniciando en 399$.

En base a esta información, se escogió como sistema operativo a Suse Linux Enterprise Server (SLES), con la principal razón de que éste cuenta con un periodo de prueba gratuito de seis meses tanto del sistema operativo como de software de alta disponibilidad; adicionalmente cuando concluye el periodo de prueba, tanto el sistema operativo como el software de alta disponibilidad siguen en funcionamiento, con la única limitante de que se restringe el acceso a los repositorios oficiales de Suse, prohibiendo la actualización de los paquetes previamente instalados.

IV.1.3 Análisis sobre el lenguaje de programación PERL

Al encontrarse con la existencia de un código previamente hecho en el lenguaje PERL, se procedió a indagar sobre las características de dicho lenguaje de programación y sobre cuales otros programas eran capaces de cumplir con las mismas actividades requeridas en este proyecto, con lo que

se encontró una cantidad sustancial de candidatos a tomar en consideración, entre los cuáles se tienen:

- Ruby
- Python
- C/C++

Cada uno de estos lenguajes mencionados tienen sus fortalezas y debilidades al momento de desempeñar una función u actividad, sin embargo, las razones principales por las que se decidió optar por el lenguaje PERL son:

- Perl toma características del lenguaje C, del lenguaje interpretado bourne shell (SH), AWK, sed y Lisp, lo cual lo convirte una excelente herramienta para el rápido proceso de texto.
- Al tomar características del lenguaje bourne Shell, dispone de funciones para interactuar con el back end de los equipos, permitiendo así ejecutar comandos de consola directamente desde el script.
- Perl puede utilizar cierres con datos privados inalcanzables como objetos.
- Corre a gran velocidad y con un buen desempeño.
- Existencia de un código inconcluso con el cual se podía hacer un solo tipo de listado o modificación en el PGW, facilitando en gran medida el trabajo ya que, no se tuvo que desarrollar desde cero el código, por el contrario, se partió de lo que se tenía disponible efectuando modificaciones para lograr el objetivo deseado.

IV.2 Fase II: Investigar la operación actual de la red DIGITEL

Al desarrollar este objetivo se pudo obtener información relevante acerca del HLR/HSS y su Single SDB, con el objetivo de llevar a cabo el proyecto, por lo que se obtuvo información relevante acerca de la distribución del equipo, redundancia, red IP, puerto y credenciales para acceder de forma remota a estos.

Single SDB: Este elemento es una versión ofrecida por Huawei en la cual integra en un mismo elemento físico un HLR y un HSS bajo una base de datos de subscriptores unificada. Acorde a la arquitectura de la Corporación, se tiene para los demás elementos de la red un Single SDB (un HLR y un HSS), pero, físicamente se tiene lo siguiente por localidad:

- Caracas: CCSSDB01 – SDB el cual se encuentra activo hasta la fecha.

- Valencia: VALSDB01 – SDB que se encuentra en espera o Standby.

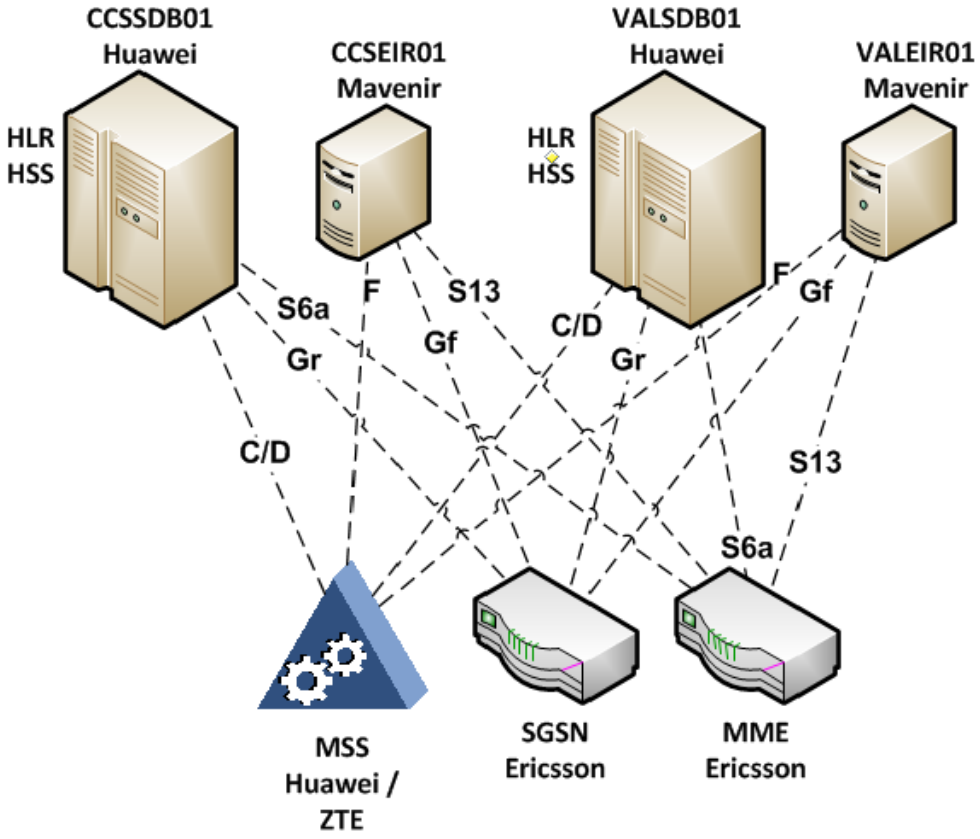


Ilustración 9. Distribución actual HLR – HSS – EIR en red Digitel

Bajo esta información, la red Digitel cuenta con 2 Single SDB bajo la estructura Activo/Standby o como se expone en el marco teórico, redundancia geográfica sin fisuras 2 x (multiples FE + BE).

IV.3 Determinar cuáles son las consultas en la HLR y HSS de mayor importancia para realizarlas a través de un programa o código desarrollado a través de un lenguaje de programación.

Para el desarrollo de esta fase se tuvo que comenzar por aprender el lenguaje de programación Perl, lo cual se logró a través de tutoriales y lecturas en la web, con el objetivo de adaptar los scripts prexistentes a las necesidades del proyecto.

Una vez asimilado el conocimiento necesario para manipular los scripts en lenguaje Perl se procedió a hacer un estudio en conjunto con el departamento para que a solicitud de ellos se dejara una lista en la que se contemplara los posibles comandos ejecutables en el SDB a través de los scripts, encontrando con los siguientes problemas:

1. La mayoría de los comandos ejecutables poseen propiedades adicionales que permiten observar o modificar información adicional de los abonados. Esto conllevo a un análisis para que el script posea la capacidad de ejecutar un comando con ninguna, una o más propiedades si el operador así lo requiera. Ej.: "LST SUB: PROPERTIES=TRUE".
2. El departamento de O&M en casos generales desconoce cuándo ocurrirá una falla, requiriendo tener a disposición una herramienta que permita ejecutar de forma masiva en sus abonados el comando que sea pertinente, atacando de forma eficaz la falla en el momento y removiendo la limitante de tener un script con un pool preconfigurado de comandos a ejecutar, dándole libertad al operador de introducir por teclado el comando especifico a ejecutar para solventar la falla.
3. Existen tres tipos de comandos ejecutables con distintos propósitos en la Single SDB, entre los que encuentran:

Grupo de comandos	Descripción
Consulta	Sirve para listar las distintas propiedades de los abonados.
Modificación	Sirve para modificar las distintas propiedades de los abonados.
Adición	Sirve para agregar propiedades
Borrado	Sirve para eliminar propiedades.

Tabla 4. Clasificación de comandos del Single SDB.

Teniendo en cuenta lo expuesto, se decidió optar por la corrección y mejora del anterior script para que los operadores de O&M Red Central Región Capital puedan a través de la terminal ejecutar el script ahora modificado e introducir por teclado el comando que se desea ejecutar de forma masiva, junto a sus propiedades si así se requiere, optimizando el alcance de este proyecto y la capacidad de respuesta del departamento.

IV.4 Diseño del sistema tentativo a implementar.

Se desarrolló un diseño del sistema deseado a implementar, basado en las necesidades y requerimientos del proyecto, pero antes de comenzar a diseñar un sistema tentativo, se procedió a enumerar y analizar las necesidades para llevar a cabo el diseño, entre las que se encuentran:

1. Redundancia entre los discos, en caso de que un disco falle, este pueda ser reemplazado en caliente y no se genere afectación en el sistema.
2. Replicación de la información alojada en los archivos de sistema.
3. Software de alta disponibilidad.
4. Redundancia entre puertos IP.
5. Un equipo externo al proyecto que pueda proveer el almacenamiento externo a través de ISCSI.

Una vez enumeradas, se procedió a analizarlas de la siguiente manera:

1. Los discos que utiliza el servidor son del tipo HDD, los cuales son muy fiables, sin embargo, el tiempo y el uso los desgasta, por lo que se deberá aplicar el arreglo de discos RAID 1 para prevenir en caso de ocurrir una falla, no ocurra perdida de información.
2. Existen varias herramientas para la replicación de los datos, sin embargo, el software de alta disponibilidad de Suse cuenta con una herramienta para esto llamada csync2.
3. Como se ha mencionado antes, se pretende usar el software de alta disponibilidad de Suse, por lo que se requerirá un disco externo a través de ISCSI para poder controlar los nodos en caso de que uno falle, este sea reiniciado.
4. La redundancia entre los puertos IP es importante en caso de que ocurra alguna falla en el cable RJ-45, en el puerto del equipo o en el puerto del Switch al que están conectados y garantizar conectividad en todo momento.
5. En el punto 3 se menciona el propósito del disco externo y gracias al departamento de O&M, se contó con el requisito.

Al contrastar las necesidades y el análisis de las mismas con los recursos con los que se disponían, se procedió a elaborar un sistema tentativo, como se puede apreciar en la ilustración 10.

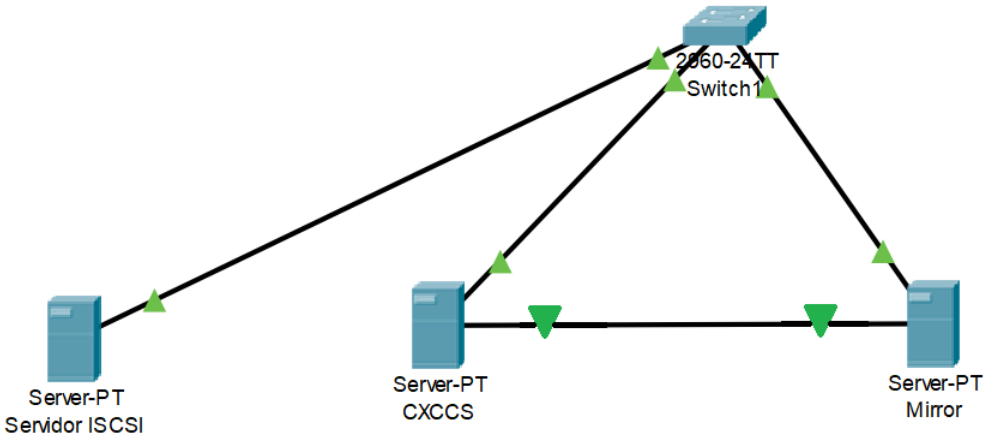


Ilustración 10. Diseño propuesto.

IV.5 Montaje y configuración afín de los servidores

En esta fase se explica en detalle los puntos que se llevaron a cabo para poner en funcionamiento el sistema a través de la configuración de los arreglos de discos, instalación del sistema operativo (OS), configuración de puertos de red, acceso al disco remoto a través de ISCSI, configuración del software de alta disponibilidad (HA) e instalación y configuración física.

IV.5.1 Arreglo de discos

En esta fase se procedió a montar y configurar el diseño que se tenía como tentativo, comenzando por la configuración de la matriz de discos RAID 1 a través de la Web BIOS, para ello se requirió reiniciar el equipo y cuando comience la secuencia de inicio se presiona <CTRL> + <H>, apareciendo el menú de configuración de arreglo de discos.

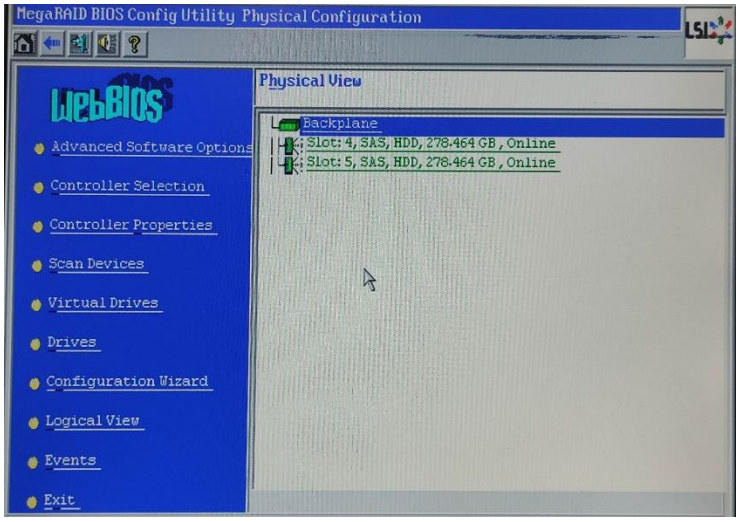


Ilustración 11. Web BIOS

Una vez en el menú inicial, se procedió a utilizar la opción *"Configuration Wizard" – del inglés, asistente de configuración para* efectuar la configuración RAID 1.

IV.5.2 Instalación del Sistema Operativo SLES 15.1

Una vez culminado el arreglo de los discos se procede a la instalación del sistema operativo, para ello se requirió grabar un CD con la imagen de Suse Linux Enterprise Server 15.1 e introducirlo en la ranura de CD/DVD del equipo y encender el mismo, a continuación, el desarrollo de la instalación:

1. Se selecciona el lenguaje y el producto a instalar.

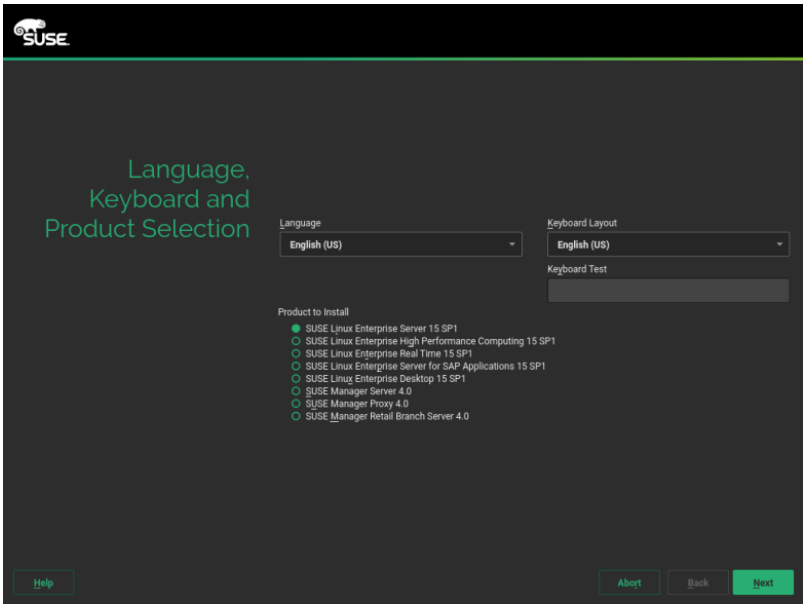


Ilustración 12. Instalación SO Suse Linux Enterprise server 15.1, selección de idioma, teclado y producto a instalar.

2. Se aceptan los acuerdos de licencia.

Ilustración 13. Acuerdo de licencia SLES.

3. Se configuran los puertos de red (IP's estáticas, DNS corporativo, Gateway y mascara) para la descarga de paquetes adicionales y poder efectuar el ingreso del serial de licencia.

(En este paso se configuró uno de los dos puertos para contar con acceso a internet con el objetivo de descargar actualizaciones).

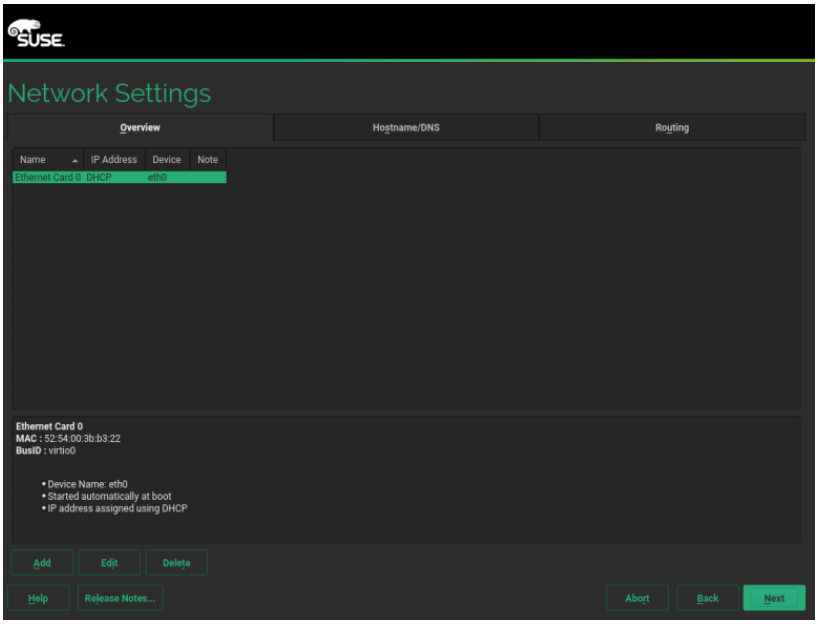


Ilustración 14. Configuración del primer puerto de red.

4. Se ingresa el correo registrado en la página de SUSE y se introduce el código de activación de prueba del producto.

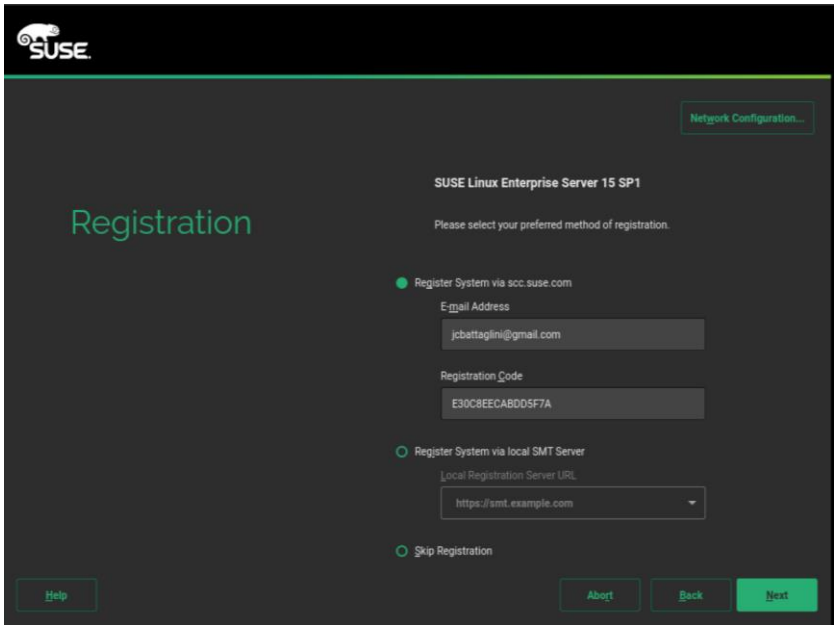


Ilustración 15. Activación del Sistema Operativo.

5. Se seleccionas las extensiones adicionales a instalar (extensión de alta disponibilidad, web server, etc.).

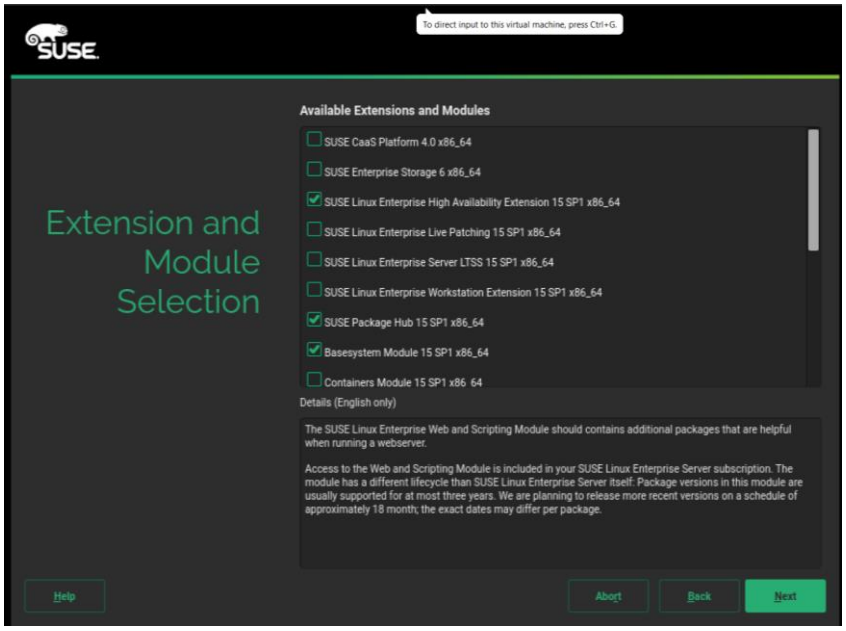


Ilustración 16. Selección de módulos a instalar.

6. Se introduce el código de registro del paquete de alta disponibilidad y se selecciona el rol del clúster (GEO clúster, nodo de alta disponibilidad o virtualización), en este caso se selecciona el nodo de alta disponibilidad.

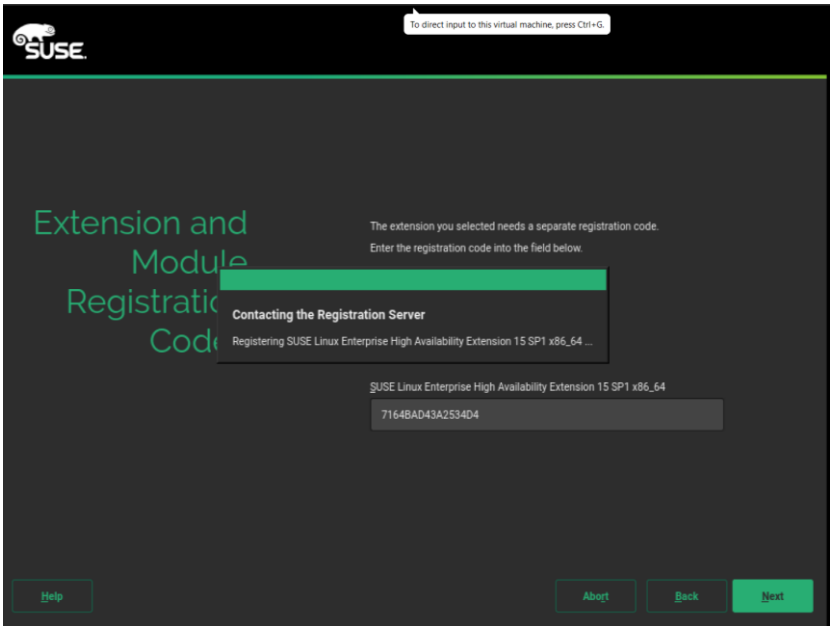


Ilustración 17. Código temporal de activación del software de Alta Disponibilidad (HA).

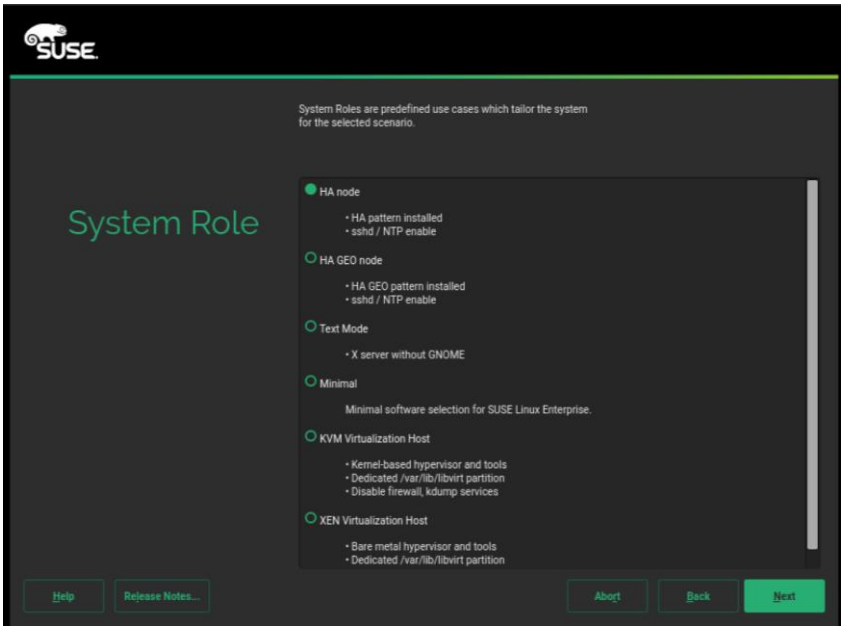


Ilustración 18. Rol del clúster.

7. Se configura el reloj y la zona horaria con el fin de usar NTP.

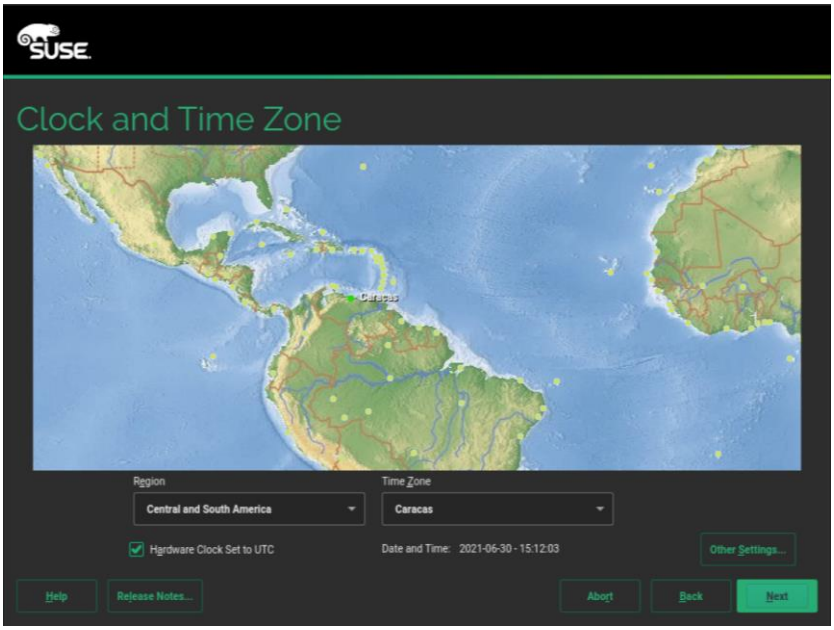


Ilustración 19. Selección de zona horaria para el uso de NTP.

8. Se configura el usuario local y contraseña de administrador del sistema o root.

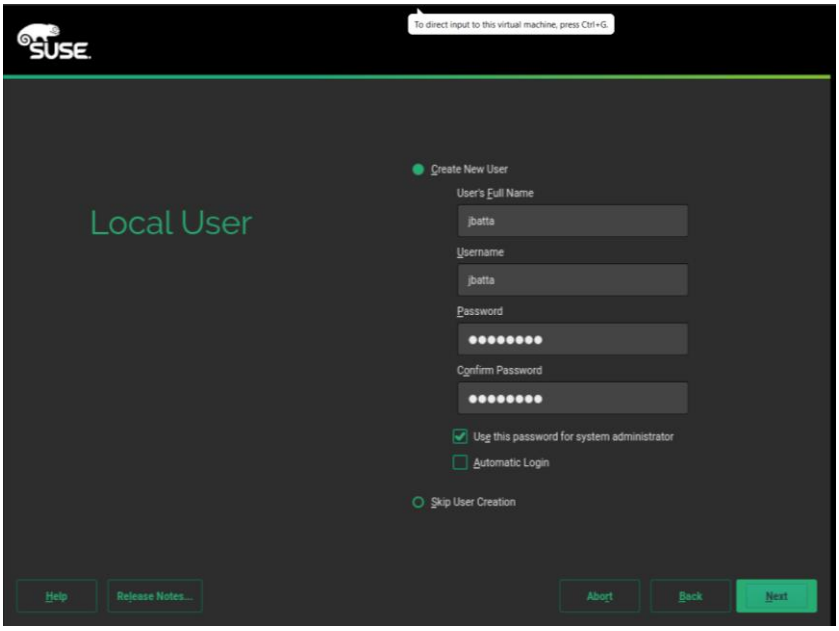


Ilustración 20. Configuración de usuario y credenciales de root.

9. Se procede a instalar el sistema operativo.

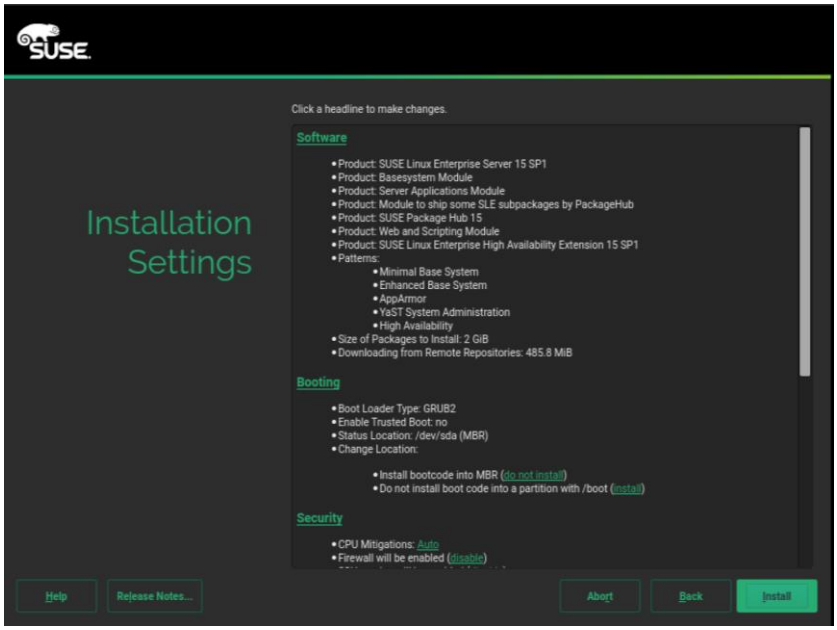


Ilustración 21. Paso final para instalar el Sistema Operativo.

Al culminar la instalación, se ingresó con las credenciales generadas en el punto 8 para efectuar las configuraciones de resolución de los nombres de hosts de los dos servidores, conexión del disco externo a través de ISCIS, actualización de paquetes, instalación de módulos de Perl y configuración del servicio de SSH para el acceso remoto.

IV.5.3 Configuración de puertos de red

Antes de llevar a cabo esta fase se solicitó al departamento de Redes dos direcciones IP's con salida a internet con el objetivo de poder ejecutar actualizaciones, descarga de paquetes afines al proyecto y contar con comunicación entre los equipos y la red de la Corporación.

Una vez asignadas las direcciones IP's solicitadas y habiendo instalado el sistema operativo, se procedió a ejecutar a través de la terminal en pantalla el comando "yast" el cual inicializa el asistente de gestión de aplicativos y herramientas de software, seleccionamos la opción de System y nos dirigimos hasta Network Settings y presionamos enter. Acto seguido aparecen en pantalla las distintas pestañas para la configuración de red como se observa en la ilustración 22, entre estas tenemos:

- Hostname/DNS: en esta parte se procede a colocar el nombre del equipo (**mirror** o **cxccs**) y agregar la dirección IP de los servidores de DNS de la corporación.

- Routing: se coloca la dirección IP de enrutamiento por la cual el equipo contara con salida a la red.
- Overview: en esta opción aparecen los puertos disponibles y la posibilidad de configurarlos con una IP estática o dinámica (DHCP), en este caso se configura dos IP's estáticas en cada servidor con sus máscaras al igual que el mtu (*mtu: máxima transfer unit, del inglés unidad de transferencia máxima*) el cual se configura a 1500.

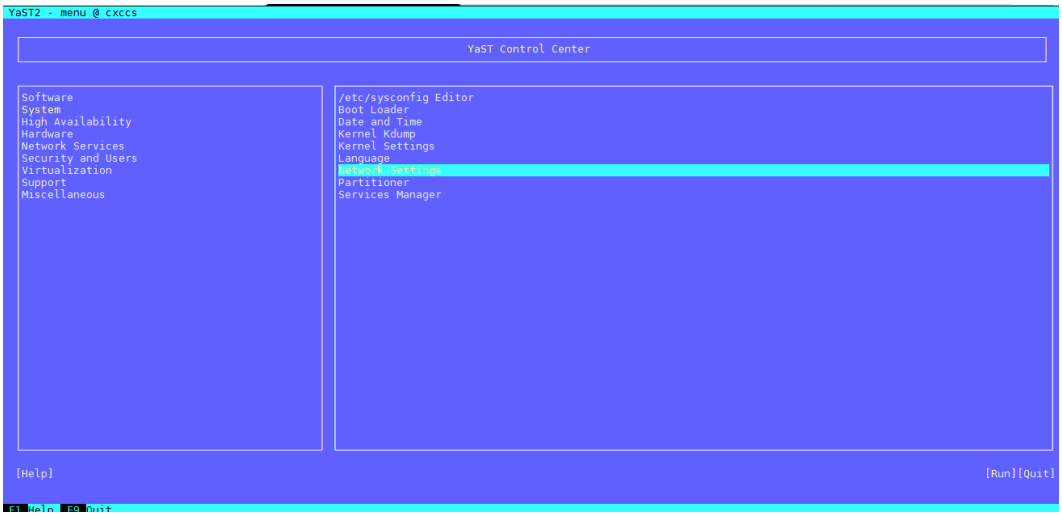


Ilustración 22. Menú asistente de configuración

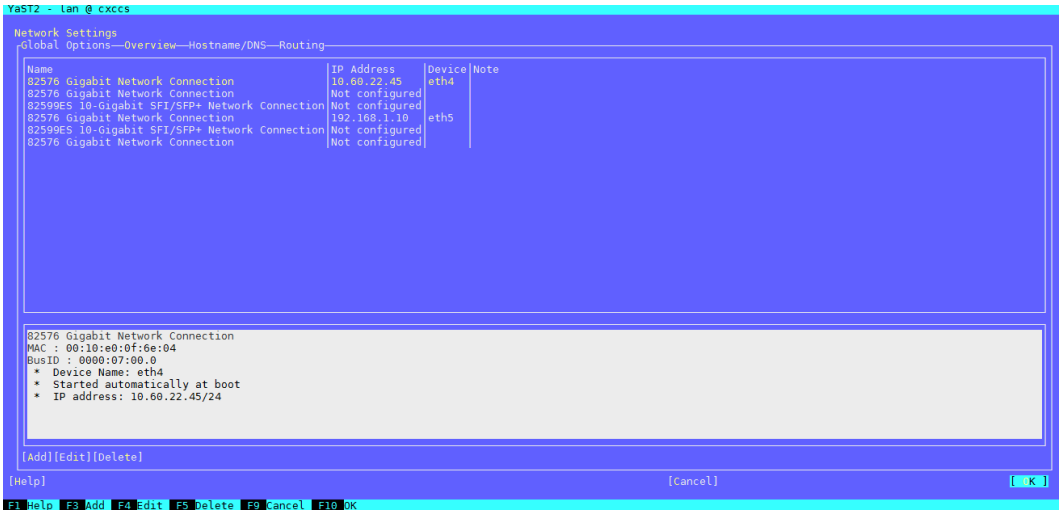


Ilustración 23. Asistente de configuración de puertos de red de servidor cxccs, también se puede apreciar los puertos configurados.

Servidor	Interfaz 1	Dirección IP/Gateway	Interfaz 2	Dirección IP/Gateway
Cxccs	Eth4	10.60.22.45/10.60.22.10	Eth5	192.168.1.10
Mirror	Eth4	192.168.1.11	Eth5	10.60.22.46/10.60.22.10

Tabla 5. Interfaces configuradas en los servidores.

IV.5.4 Configuración ISCSI

Para configurar ISCIS se adaptó un servidor externo (OS Windows server 2016) como target u objetivo, el cual proveerá el almacenamiento externo a usar por los servidores de este proyecto, para ello se instala el software en este equipo y se configura el nombre del target y el almacenamiento dispuesto a ceder para este servicio, en este caso son dos secciones de 250 MB, posteriormente se configura el servicio ISCSI como iniciador en los servidores del proyecto, a continuación se detallan las configuraciones efectuadas.

IV.5.4.1 Servicio ISCSI como Objetivo

La configuración de este servicio fue requerido el acceso a un servidor del departamento de O&M el cual cuenta con un sistema operativo Windows server 2016 e IP 10.60.22.38 a través del puerto 53260 y llevar a cabo los siguientes pasos:

1. Abrir el Administrador del Servidor, hacer abrir la opción “administrar" y seleccionar la opción de “agregar roles y características”.

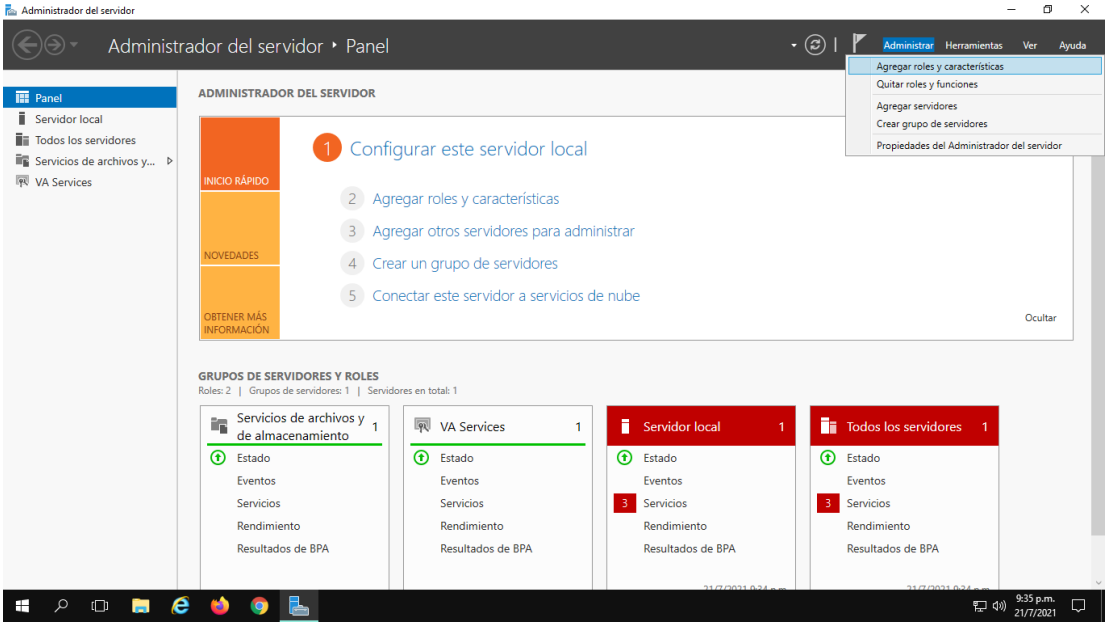


Ilustración 24. Instalación Servicio ISCSI como Objetivo paso 1.

2. Seleccionar el rol del servicio a instalar.

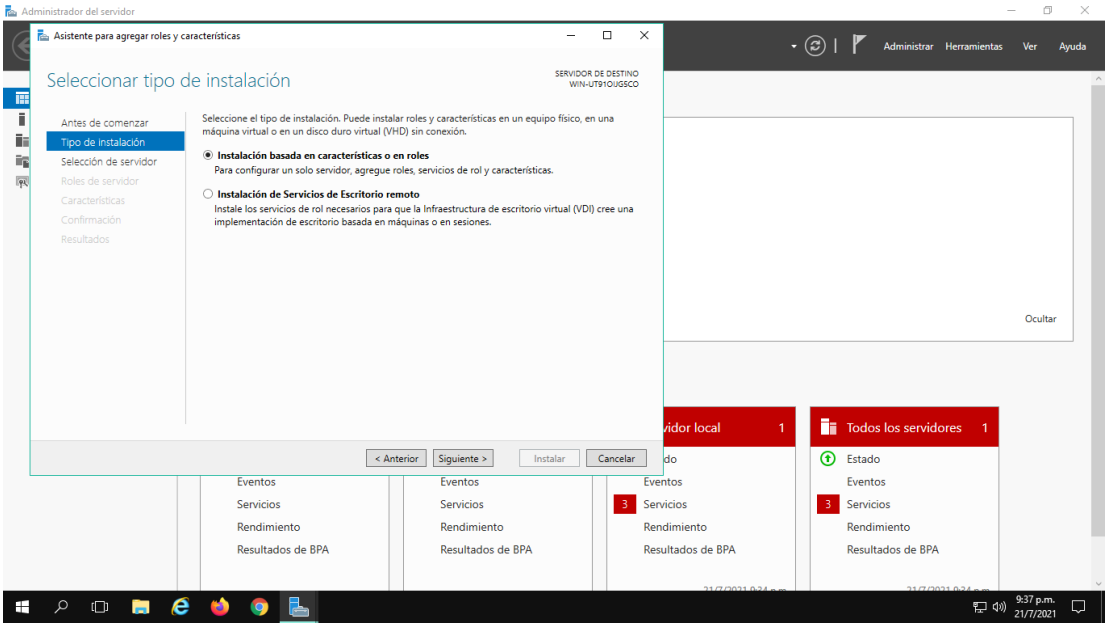


Ilustración 25. Instalación Servicio ISCSI como Objetivo paso 2.

3. Seleccionar la opción “Seleccionar un servidor del grupo de servidores”.

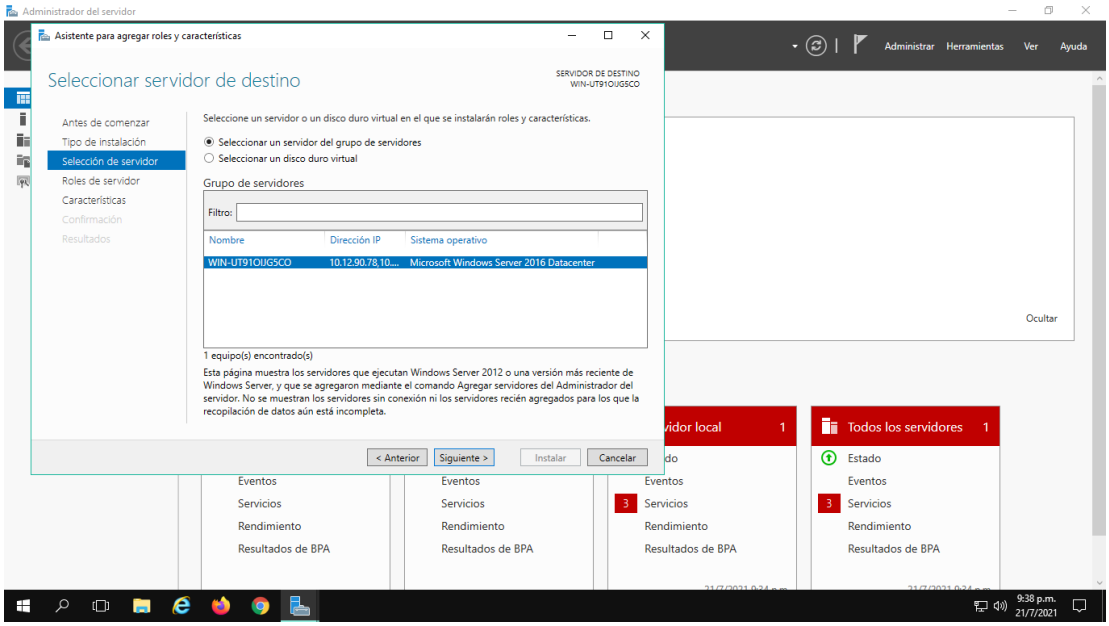


Ilustración 26. Instalación Servicio ISCSI como Objetivo paso 3.

4. Seleccionar el rol del servidor, en este caso “Servicios de archivo y almacenamiento”.

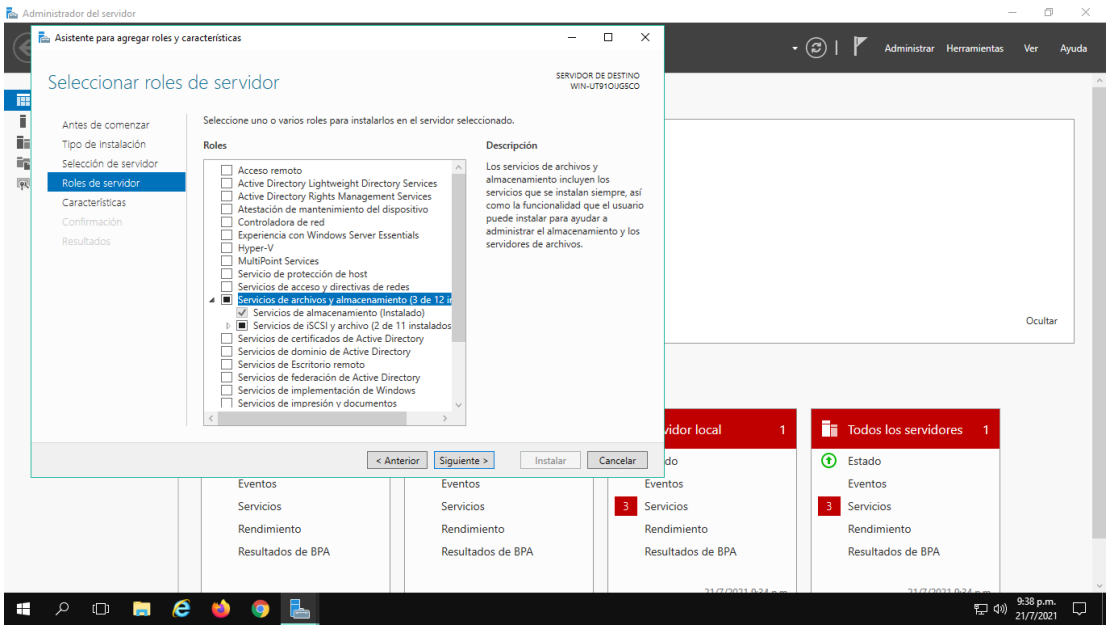


Ilustración 27. Instalación Servicio ISCSI como Objetivo paso 4.

5. Se selecciona “Características de .NET Framework”

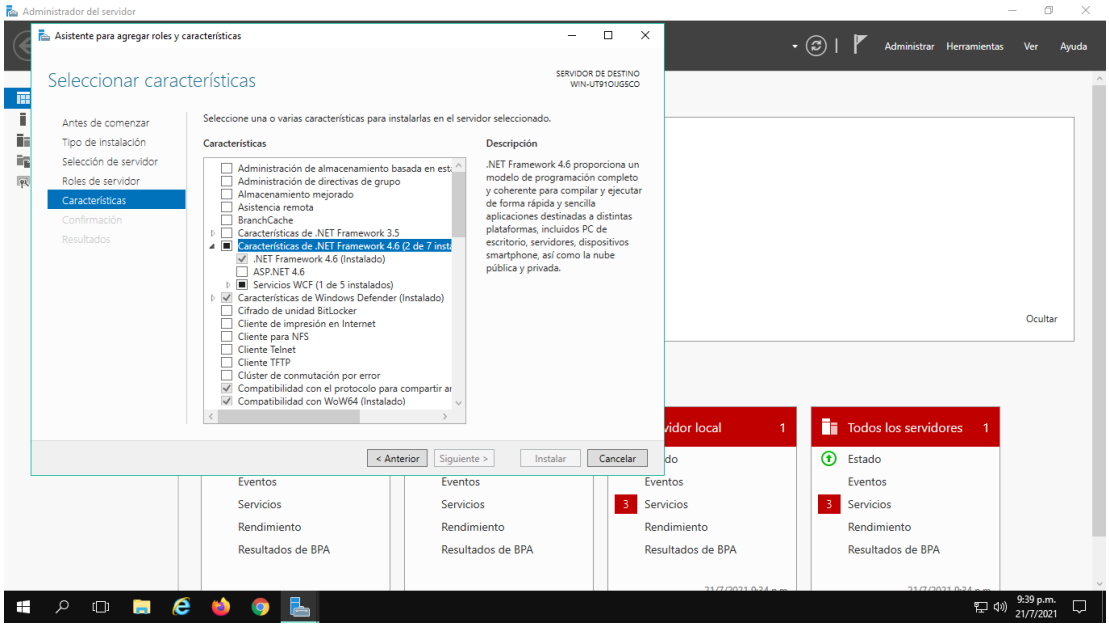


Ilustración 28. Instalación Servicio ISCSI como Objetivo paso 5.

6. Se selecciona la viñeta de “Reiniciar el servidor de destino automáticamente si es requerido”.

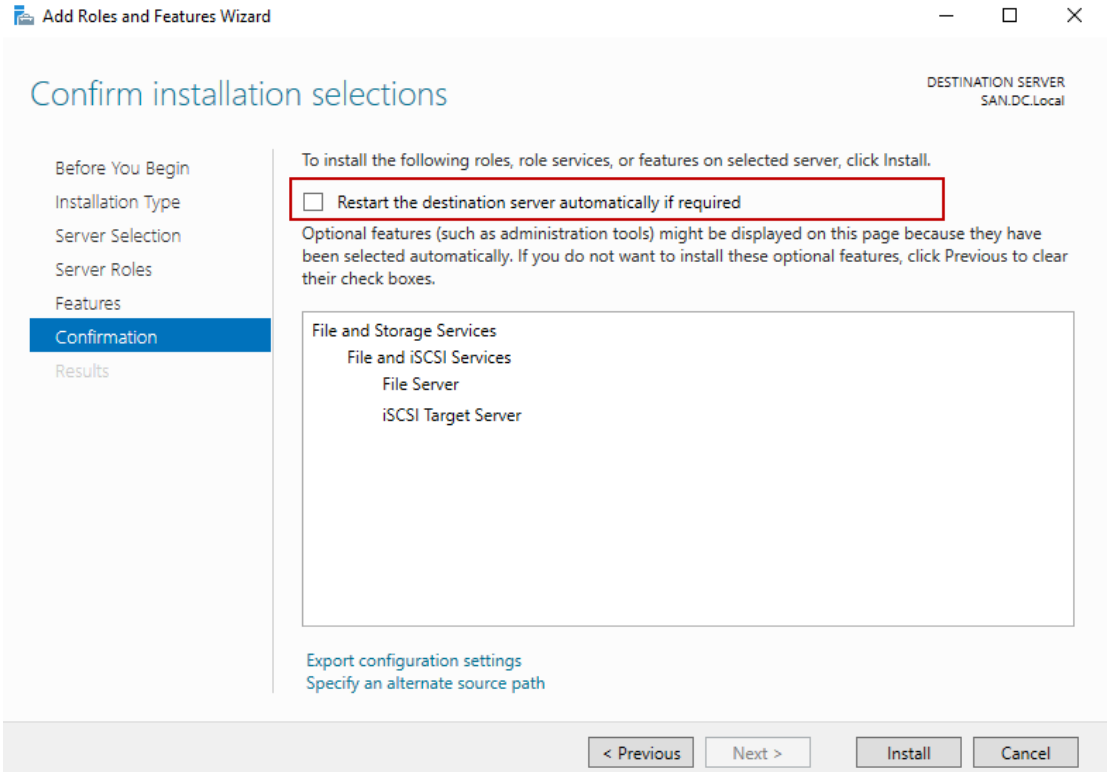


Ilustración 29. Instalación Servicio ISCSI como Objetivo paso 6.

7. Por último se selecciona la opción de “Instalar”

IV.5.4.2 Servicio ISCSI como Iniciador

Para llevar a cabo la configuración de ISCSI como iniciador en los servidores se ejecuta a través de terminal el comando “yast”, se procede a servicios de red y se selecciona la opción “ICSI Initiator”.

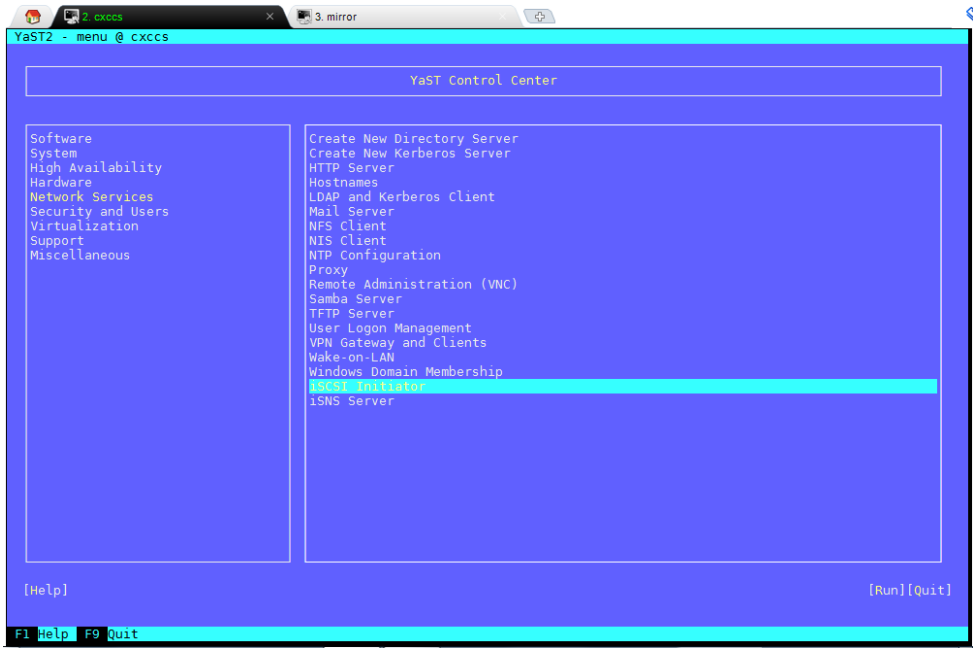


Ilustración 30. Menú del centro de control de servicios de red.

Una vez dentro se procede a la pestaña "Discovered Targets", se selecciona la opción "Discovery" y se introduce la dirección IP del target "10.60.22.38", de esta manera aparecerá dentro de la lista de objetivos descubiertos y se procede a conectarse a este.

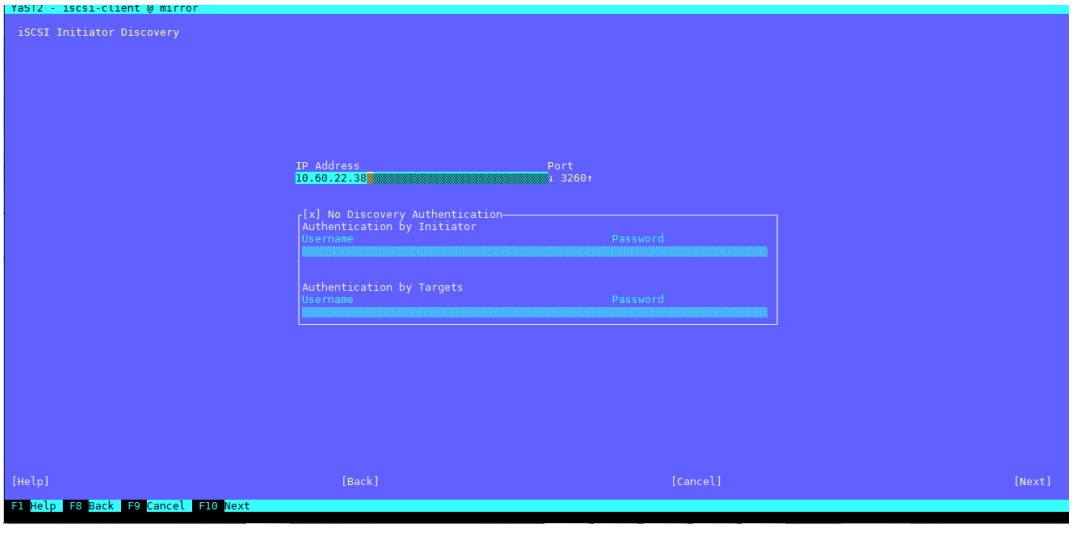


Ilustración 31. Asistente de configuración ISCSI en el que podemos apreciar la IP y puerto del Target o servidor objetivo.

IV.5.5 Configuración del software de Alta Disponibilidad (HA)

Para configurar el software de alta disponibilidad se requirió configurar previamente un conjunto de parámetros mínimos o de lo contrario el software no desempeñaría sus funciones de manera adecuada, a continuación, se explican las configuraciones efectuadas.

IV5.5.1 Configuración SBD

Para comenzar con las configuraciones mínimas del clúster fue necesaria la configuración del mecanismo de cercado de nodos mediante el uso de SBD utilizando como bloque de memoria las particiones configuradas a través de ISCSI, para ello se debió editar el archivo /etc/sysconfig/sbd, agregar el nombre y ruta estable y persistente (en ambos nodos) de las particiones en la línea que comienza con "SBD_DEVICE=" y agregar ambas rutas como aparece en la ilustración 32. Cabe destacar que el nombre de la ruta para ser estable debe de ser del tipo como se muestra en el siguiente ejemplo: /dev/disk/by-id/dm-uuid-part1-mpath-abcedf12345.

Para editar el archivo ejecutamos el comando "vi" seguido de la ruta, de la siguiente forma "vi /etc/sysconfig/sbd" y una vez abierto se edita.

Ilustración 32. Edición de rutas consistentes y estables obtenidas a través de la configuración del servicio ISCSI y agregadas al demonio SBD.

IV.5.5.2 Configuración Watchdogtime

La configuración de este componente fue indispensable ya que el software de alta disponibilidad utiliza el demonio SBD como el componente de software que alimenta al Watchdog. SBD necesita que ambos nodos tengan Watchdog para asegurar que los nodos que fallen sean detenidos.

Para configurar el uso del Watchdog se ejecutaron los siguientes comandos:

- Para cargar el módulo a utilizar: "echo softdog > /etc/modules-load.d/watchdog.conf "
- Para reiniciar el servicio y actualizar los cambios efectuados "systemctl restart systemd-modules-load"
- Para constatar que el módulo fue cargado correctamente "lsmod | grep dog"

IV.5.5.3 Configuración del primer nodo

Para configurar el primer nodo se ejecutó el script "ha-cluster-init –name", requiriendo un mínimo de tiempo e intervención manual, a continuación, el procedimiento:

1) Ingresar como root a la maquina física.
2) Ejecutar el script de Bootstrap ejecutando: "ha-cluster-init --name CLUSTERNAME". Reemplazar CLUSTERNAME con el nombre a asignar al clúster. Este script verifica la existencia de la configuración de NTP y el servicio de watchdog. Genera las llaves públicas y privadas de SSH y la sincronización a través de Csync2 e inicializa los respectivos servicios.

```
group ha_group
{
        key /etc/csync2/key_hagroup;
        host mirror;
        host cxccs;
        include /etc/booth;
        include /etc/corosync/corosync.conf;
        include /etc/corosync/authkey;
        include /etc/csync2/csync2.cfg;
        include /etc/csync2/key_hagroup;
        include /etc/ctdb/nodes;
        include /etc/drbd.conf;
        include /etc/drbd.d;
        include /etc/ha.d/ldirectord.cf;
        include /etc/lvm/lvm.conf;
        include /etc/multipath.conf;
        include /etc/samba/smb.conf;
        include /etc/sysconfig/pacemaker;
        include /etc/sysconfig/sbd;
        include /etc/pacemaker/authkeyo;
        include /srv/www/htdocs/PERL/PGWBatch.pl;
        include /srv/www/htdocs/PERL/123.pl;
        include /srv/www/htdocs/PERL/bb.pl;
        include /srv/www/htdocs/PERL/filtro.pl;
        include /srv/www/htdocs/PERL/CCSPGW01.porperties;
}
```

Ilustración 33. Archivo de configuración de Csync2

3) Configuración de la capa de comunicación del clúster (Corosync):
   a) Se ingresó una dirección de red a la cual enlazar y se utilizó una dirección IP de multicast, en este caso la IP 239.163.19.247 a través del puerto 5405 y 239.163.19.246 a través del puerto 5406.

4) Establecer SBD como mecanismo de cercado de nodos.
   a) Confirmar con una "y" afirmando que se desea utilizar SBD.
   b) Ingresar el camino persistente a la partición del dispositivo de bloques que se desea usar para SBD (en este caso la ruta de los discos utilizados a través de ISCSI). El camino debe ser consistente en todos los nodos del clúster. En este caso ya se había configurado el demonio SBD por lo que aparecen los caminos ya configurados, sin embargo, el script de Bootstrap solicitara confirmar si se desea usar la configuración existente o si se requiere usar una nueva.
5) Configurar una IP virtual para la administración del clúster utilizando Hawk2.
   a) Confirmar con una "y" que se desea configurar la IP virtual.
   b) Ingresar una dirección IP disponible dentro de la red local que se desee utilizar como IP administrativa para Hawk2, en este caso 10.60.22.250/24.

Finalmente, el script iniciara el servicio Pacemaker para poner el clúster en funcionamiento y habilitar Hawk2. El URL de Hawk2 aparecerá en pantalla o, en su defecto usamos la IP virtual seguida por dos puntos y el puerto como se muestra a continuación: https://10.60.22.250:7630/

IV.5.5.4 Configuración del segundo nodo

En la configuración del segundo nodo se ejecutó el comando: "ha-cluster-join", cabe destacar que el script de Bootstrap se encarga de cambiar la configuración especifica de un clúster de dos nodos por ejemplo en los servicios de SBD y Corosync. A continuación, el procedimiento ejecutado:

1. Ingresar como root en la maquina física a agregar al clúster.
2. Inicializamos el script de Bootstrap ejecutando "ha-cluster-join". Este script verifica la existencia de la configuración de NTP.
3. Al continuar se solicitará la dirección IP de un nodo existente. Se ingresamos la IP del primer nodo (cxccs, 10.60.22.45).

IV.5.6 Instalación y configuración física

Para llevar a cabo la instalación física de los equipos se llevó a cabo en primera instancia una visita a la sede de la Corporación Digitel ubicada en Sartenejas con los siguientes objetivos:

1. Verificar la ubicación propuesta para instalar los equipos.
2. Corroborar disponibilidad de espacio en el rack de servidores.
3. Disponibilidad de puertos en el switch a través del cual se tendrá acceso a la red de la Corporación y salida a internet en caso de ser necesario.

4. Disponibilidad de tomas de energía para las fuentes activa/respaldo, ya que se requiere garantizar la operatividad del sistema en caso de fallar una de las fuentes o en caso de falta de suministro eléctrico debido a un corte de energía en la zona.
5. Instalación de los rieles.
6. Instalación de los servidores, energizar los equipos y conectarlos al switch.

IV.6 Pruebas de funcionamiento y operatividad del servidor.

En esta fase se llevó a cabo un conjunto de pruebas con el objetivo de detectar fallas y mal funcionamientos para poder ejecutar correcciones, apunte de observaciones y análisis en base a los resultados obtenidos.

Entre las pruebas efectuadas se tienen:

IV.6.1 Prueba de discos

Esta prueba tuvo como propósito constatar la adecuada configuración del arreglo matricial de discos (RAID 1). Para ello se efectuó la extracción de uno de los discos duros en caliente que posee cada servidor y reinsertarlos. Si durante esta prueba el servidor continúa operando con normalidad se considera exitosa.

IV.6.2 Redundancia de puertos

Esta prueba se llevó a cabo con la finalidad de verificar si en efecto se logra establecer comunicación entre los servidores a través de ambas salidas fast ethernet configuradas y constatar la presencia de redundancia. Esta prueba se desarrolla a través de la ejecución del comando “ping” entre el servidor “A” a ambas direcciones IP del servidor “B” y viceversa y, se considera exitosa si en efecto se envían y reciben los paquetes a través del comando previamente mencionado.

IV.6.3 Resolución de Hostname o nombre del hospedador (servidor)

Esta prueba se dispuso para constatar que, al igual que la prueba anterior existiese comunicación entre los servidores solo que en este caso no se utiliza la dirección IP de los equipos, en vez se utiliza el nombre del equipo con el que se desea entablar comunicación y, al igual que el caso anterior se realiza a través de la ejecución del comando “ping” entre ambos servidores y se considera la prueba exitosa si se envían y reciben los paquetes de datos.

IV.6.4 Acceso remoto

Esta prueba fue crucial, ya que, el acceso a los equipos se tiene planeado que sea 100% remoto a través del protocolo TCP/IP utilizando el servicio de SSH a través del puerto 22. Esta prueba se considera exitosa si desde un equipo remoto que cuente con acceso a la red de la CORPORACION DIGITEL y tenga instalado un software de SSH (ej. Putty o Moba) pueda acceder a los servidores ingresando nombre de usuario y contraseña.

IV.6.5 Acceso remoto a través de la IP virtual

Esta prueba tiene como finalidad corroborar el funcionamiento de la IP virtual que será configurada a través del software de alta disponibilidad y al igual que la prueba anterior a esta, se constata a través del uso de un software de SSH ingresando la dirección IP virtual configurada e ingresando las debidas credenciales. También se puede corroborar al ingresar a través de la interfaz web en donde se observa en la pestaña de recursos que la IP virtual tiene estatus activo.

IV.6.6 Reinicio o cercado del nodo

Esta prueba pretende constatar el correcto funcionamiento del software de alta disponibilidad al momento de reiniciar uno de los nodos debido a una falla o por problemas de conexión. Posterior al ciclo de reinicio, los servicios deben ser migrados automáticamente y, una vez reiniciado el nodo, éste debe de ponerse en espera o standby.

Cabe destacar que en la ejecución de estas pruebas se encontraron dos problemas:

1. Al reiniciar los servidores se observó de forma reiterada complicaciones para arrancar el sistema operativo, encontrando problemas relacionados al CMOS debido a que la batería que lo alimenta se encontraba descargada. La solución fue el cambio de ésta en ambos equipos, para ello se tuvo que desinstalar del rack los equipos, remover la tapa superior y efectuar el cambio.

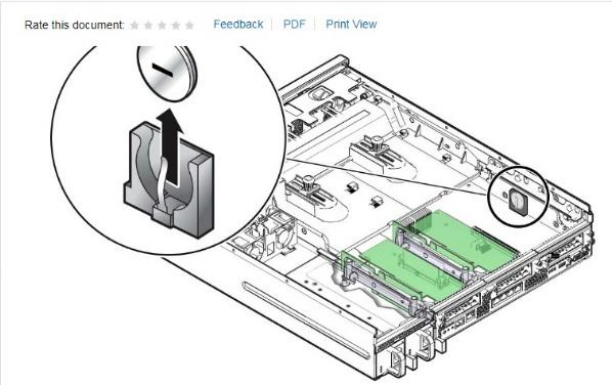


Ilustración 34. Manual para el reemplazo de la batería del CMOS.

Ilustración 35. Batería BR 2032.

2. Cuando uno de los nodos culmina el ciclo de reinicio, éste debe reintegrarse al clúster en modo de espera, sin embargo, esto no ocurría, obligando a agregarlo de forma manual mediante la ejecución de comandos en la interfaz de comandos de línea (CLI) de SUSE. Para corregir esto, se generó un script en el que se ejecutan los comandos de línea necesarios para reintegrar el servidor (que fue reiniciado) al clúster. El script es ejecutado de forma automática durante el ciclo de encendido mediante el uso de la herramienta CRONTAB, la cual permite ejecutar comandos de línea durante el ciclo de encendido.

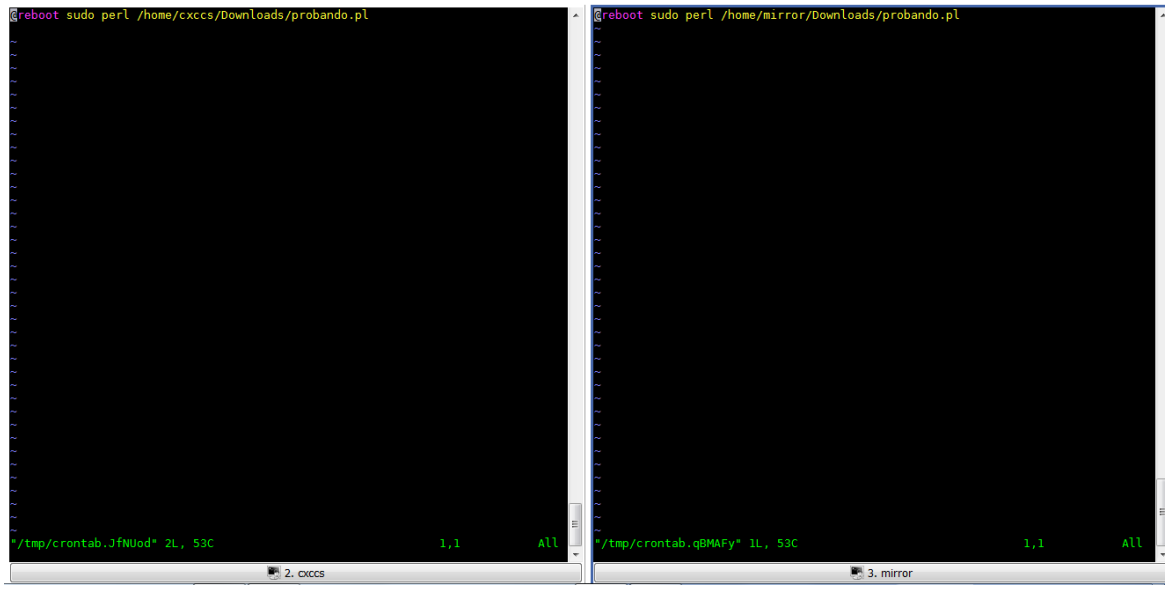


Ilustración 36. Script a ejecutar a través del crontab.

IV.6.7 Ejecución de Scripts con lista de abonados

Esta prueba fue una de las más extensas ya que requirió la ejecución de los scripts con diferentes cantidades de abonados para constatar el comportamiento de los scripts, efectuar correcciones en los mismos para que su uso sea lo más ameno posible y observar el tiempo requerido de ejecución con las distintas cantidades de abonados y con un comando de listado (simple y extendido). En este caso se llevó a cabo la ejecución con 1.000, 5.000, 10.000, 50.000 y 100.000 abonados en horario diurno y nocturno mediante el uso del comando de listado LST SUB y de listado con propiedades extendidas “LST SUB” “PROPERTIES=TRUE”.

Entre los problemas encontrados en su totalidad fueron de sintaxis o de uso indebido de condicionales o de ciclos los cuales se solucionaron mediante una inmersión profunda en el lenguaje

Capítulo V

Resultados

En este capítulo se exponen los resultados obtenidos a partir del desarrollo de todas las fases del Trabajo Especial de Grado. Se enfoca en la edición de los scripts de programación para el uso del departamento de Operación y mantenimiento (O&M); Implementación del diseño propuesto; Instalación física de los servidores; Pruebas de operatividad y funcionamiento del proyecto. Todo esto con el objetivo de evaluar el proyecto, su alcance, limitaciones y recomendaciones para su uso dentro de la Corporación Digitel.

V.1 Investigar antecedentes bibliográficos y tecnologías sobre el tema y la red Digitel.

Se investigó una base teórica sustancial que permite entender, configurar, manipular e implementar cada uno de los objetivos a desarrollar a lo largo de este Proyecto Especial de Grado junto con su debida documentación, con excepción al desarrollo de los scripts de programación.

V.2 Investigar la operación actual de la red DIGITEL

En este objetivo se investigó la información necesaria para entablar comunicación entre los servidores del proyecto y el Single SDB mediante el uso de los scripts de programación en lenguaje Perl. Entre la información recabada se obtuvo la dirección IP, puerto y servicio de red utilizados para entablar comunicación con en Single SDB que por motivos de seguridad no se expondrán en este proyecto.

V.3 Determinar cuáles son las consultas en la HLR y HSS de mayor importancia para realizarlas a través de un programa o código desarrollado a través de un lenguaje de programación.

Como se expone en el capítulo IV.3, se determinó que el proyecto tiene un mayor alcance si en vez de preconfigurar un pool determinado de comandos ejecutables, (que no cubre en su totalidad la posibilidad de comandos dentro del Single SDB) se opta por darle libertad al operador de ejecutar el comando que desee a través de la terminal introduciéndolo por teclado junto con propiedades adicionales

si así lo desea. Esto se logra mediante la modificación del script, haciendo que tome cadenas de caracteres introducidas mediante teclado. En la siguiente imagen, se aprecia cómo se ejecuta el programa junto con el comando seguido de una propiedad.

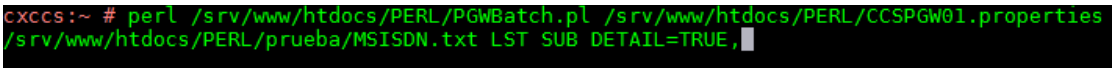


Ilustración 37. Ejemplificación de cómo se ejecutan los scripts junto al respectivo comando a correr en la Single SDB.

Esto es posible, ya que, Perl le permite al usuario interactuar con el programa al introducir argumentos, cada uno separado por espacio, en la misma línea de comandos. Cada argumento introducido es tomado dentro del arreglo matricial especial ARGV (ARGV[0], ARGV[1], ARGV[2],…), conteniendo cada variable para el funcionamiento del script acorde al orden de entrada. Por ejemplo, en la imagen anterior se observa que ARGV almacenará en su matriz especial todo lo escrito después del nombre del script ejecutado, a continuación, una tabla explicativa:

Título	Argumentos	Variables
Comando	Perl	
Archivo Perl	/srv/www/htdocs/PERL/PGWBatch.pl	
Propiedades	/srv/www/htdocs/PERL/CCSPGW01.properties	$ARGV[0]
Lista MSISDN	/srv/www/htdocs/PERL/prueba/MSISDN.txt	$ARGV[1]
Tipo de Comando	"LST SUB"	$ARGV[2]
Comando específico	SUB	$ARGV[3]
Propiedades adicionales	DETAIL=TRUE	$ARGV[4]

Tabla 6. Comando especial $ARGV.

Como $ARGV[0] y $ARGV[1] no suelen variar, se pasan valores a $ARGV[2], $ARGV[3] y $ARGV[4] para que el script ejecute sus comandos en el Single SDB, como se puede observar en la siguiente imagen. Se resalta que ARGV logra esto al discriminar la separación por espacios.

```
171
172     foreach $iNumber (@iArray) {
173         $iNumber = &Trim ($iNumber);
174         if ($iNumber !~ /^58[4]\d{9}$/) {&PrintOutput ("Error: The IMSI ($iNumber) is incorrect."); next;}
175
176         # Extraer ODBs y ejecutar cambios si aplican
177         $prematch = &HExecuteCommand ("$ARGV[3] $ARGV[4]: $ARGV[5] ISDN=\"$iNumber\";", $properties->getProperty("\QPrompt_Initial_Command"),
178             $properties->getProperty("General_Timeout"));
179         if (uc ($prematch) =~ /^ERR\d+[:](?^A.*)$/gm) {
180             &PrintOutput ("Error:$+{A},MSISDN=$iNumber"); next;
181
182     }
183
184 }
185
```

Ilustración 38. Resultado de la adaptación del script para la ejecución de comandos ingresados por teclado.

V.4 Diseño del sistema implementado.

Se cumplió con este objetivo al trazar un diseño definitivo a ser implementado, partiendo por los requerimientos necesarios para su implementación hasta llegar a la instalación del sistema, dichos requisitos están debidamente documentados en este Proyecto Especial de Grado, partiendo por obtener información relevante del Single SDB, edición de los scripts de programación, arreglo de discos en RAID 1, Instalación del sistema operativo en ambos servidores, configuraciones de red y de los servicios necesarios para la implementación del clúster de alta disponibilidad a través del software de Suse. A continuación, se muestra el diseño final.

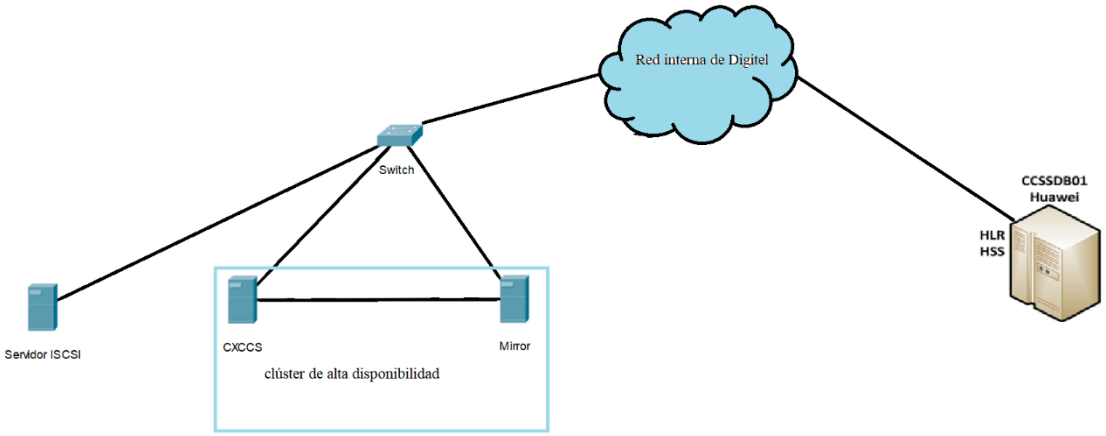


Ilustración 39. Diseño final del sistema.

V.5 Montaje y configuración afín de los servidores

Se cumplió este objetivo al lograr las configuraciones necesarias para la implementación de un sistema capaz de satisfacer las necesidades de este proyecto. A continuación, se exponen los resultados obtenidos a lo largo del desarrollo de esta fase:

V.5.1 Arreglo de discos

Los resultados obtenidos al configurar el arreglo de discos RAID 1 a través de la Web BIOS se pueden constatar en las siguientes figuras.

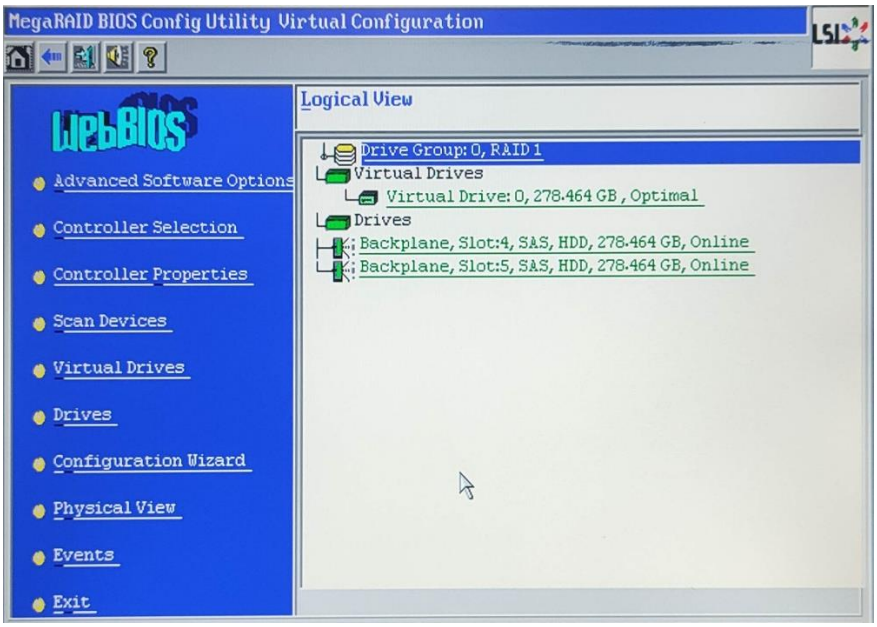


Ilustración 40. Arreglo Raid 1 a través de la Web BIOS.

```
cxccs:~ # lsblk
NAME                                  MAJ:MIN RM   SIZE RO TYPE  MOUNTPOINT
loop0                                   7:0    0  99.4M  1 loop  /snap/core/11316
loop1                                   7:1    0  99.4M  1 loop  /snap/core/11187
loop2                                   7:2    0 280.2M  1 loop  /snap/vuze-vs/3
sda                                     8:0    0 278.5G  0 disk
├─sda1                                  8:1    0     8M  0 part
└─sda2                                  8:2    0 278.5G  0 part
  ├─system-swap                       254:0    0  47.2G  0 lvm   [SWAP]
  ├─system-home                       254:1    0   155G  0 lvm   /
  └─system-root                       254:2    0  52.2G  0 lvm
sdb                                     8:16   0    25G  0 disk
└─360003ff44dc75adcaf74a6985dcfb64f 254:3    0    25G  0 mpath
  └─vg_apache-lv_apache               254:6    0    25G  0 lvm
sdc                                     8:32   0   250M  0 disk
└─360003ff44dc75adcbfd380185696286c 254:4    0   250M  0 mpath
sdd                                     8:48   0   250M  0 disk
└─360003ff44dc75adc98bd47a85c378c4d 254:5    0   250M  0 mpath
sr0                                    11:0    1  1024M  0 rom
cxccs:~ #
```

Ilustración 41. Resultado del arreglo de discos en Raid 1.

En la figura 40 se observar que sólo aparece un disco de 278.5 GB, esto se debe a que en la Web BIOS se configuró el “Drive Group:0, Raid 1” por lo que el sistema operativo ve esta agrupación de discos como un sólo disco.

V.5.2 Instalación del Sistema Operativo SLES 15.1

Este requerimiento se cumplió al finalizar los pasos necesarios para la instalación del sistema operativo Suse Linux Enterprise Server 15.1, a continuación, se evidencian los resultados.

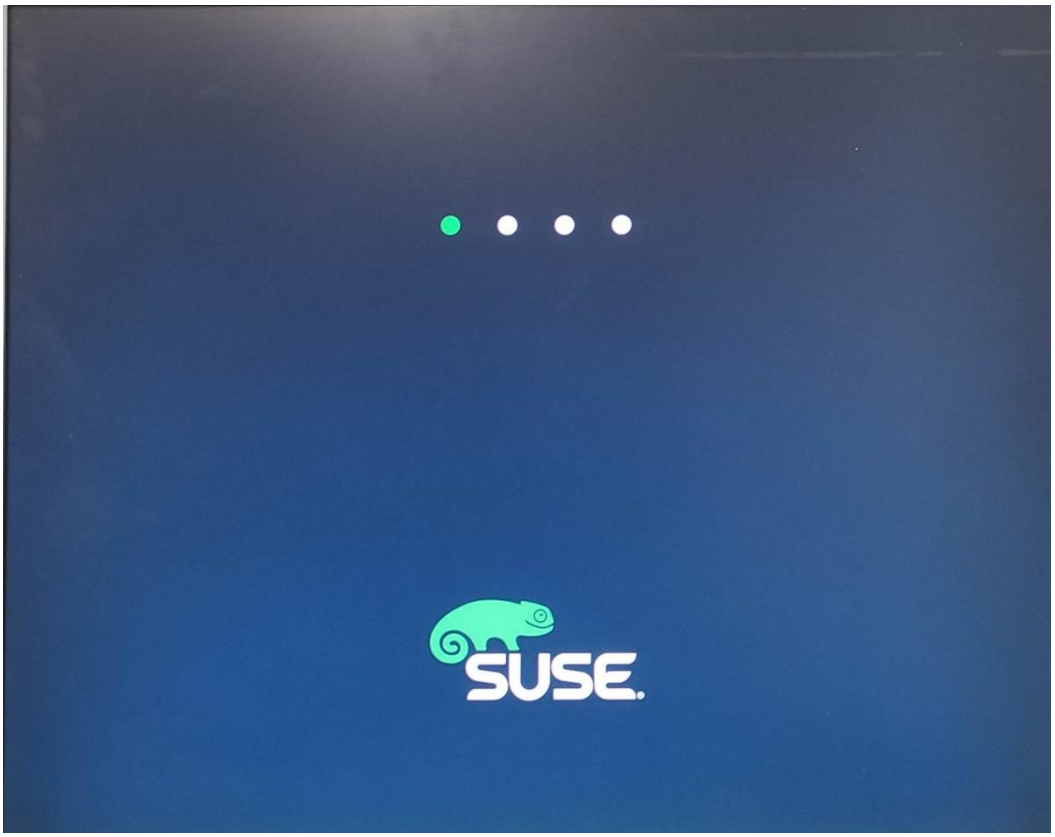


Ilustración 42. Sistema operativo iniciando por primera vez.

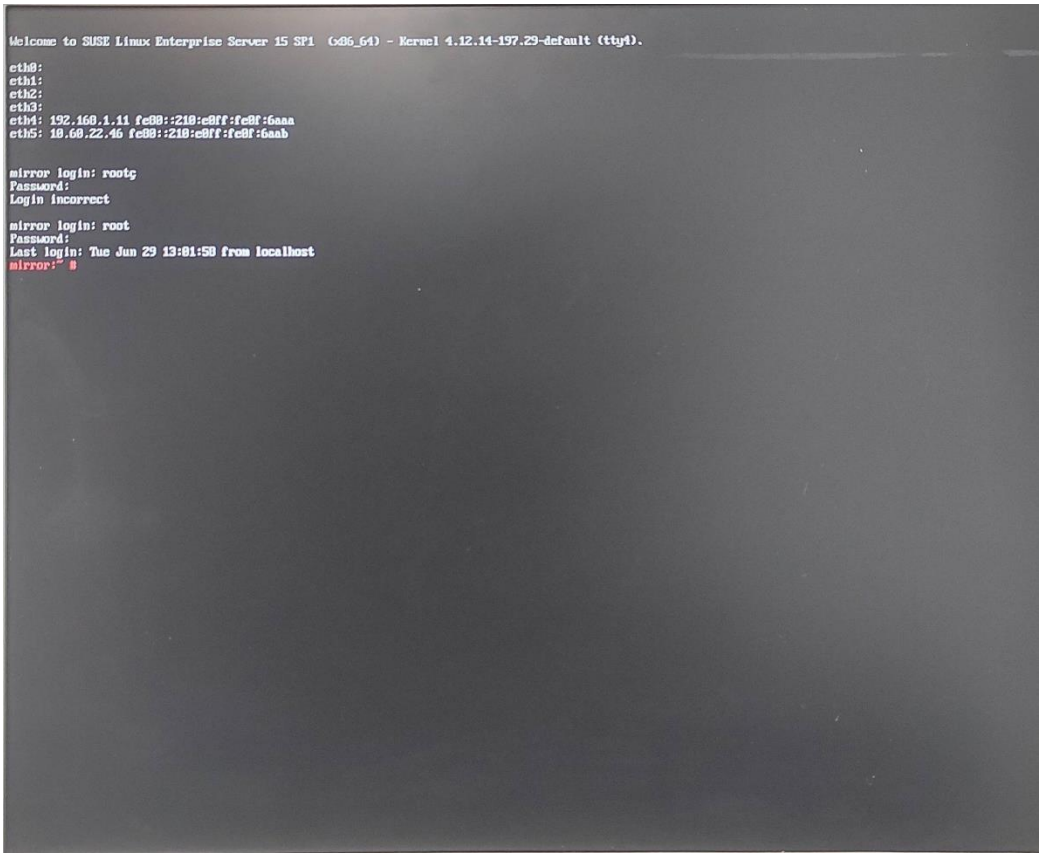


Ilustración 43. Inicio de sesión en el Sistema Operativo SLES 15.1.

V.5.3 Configuración de puertos de red

Esta configuración se satisfizo al lograr configurar los puertos de red con a través de los cuales el sistema emprende comunicación con la red interna de la Corporación Digitel, requisito indispensable si se desea manipular el Single SDB, tener comunicación entre los servidores, obtener acceso al sistema a través del servicio de SSH, acceso a los bloques de memoria configurados a través del servicio ISCSI para el cercado de los nodos, entre otros.

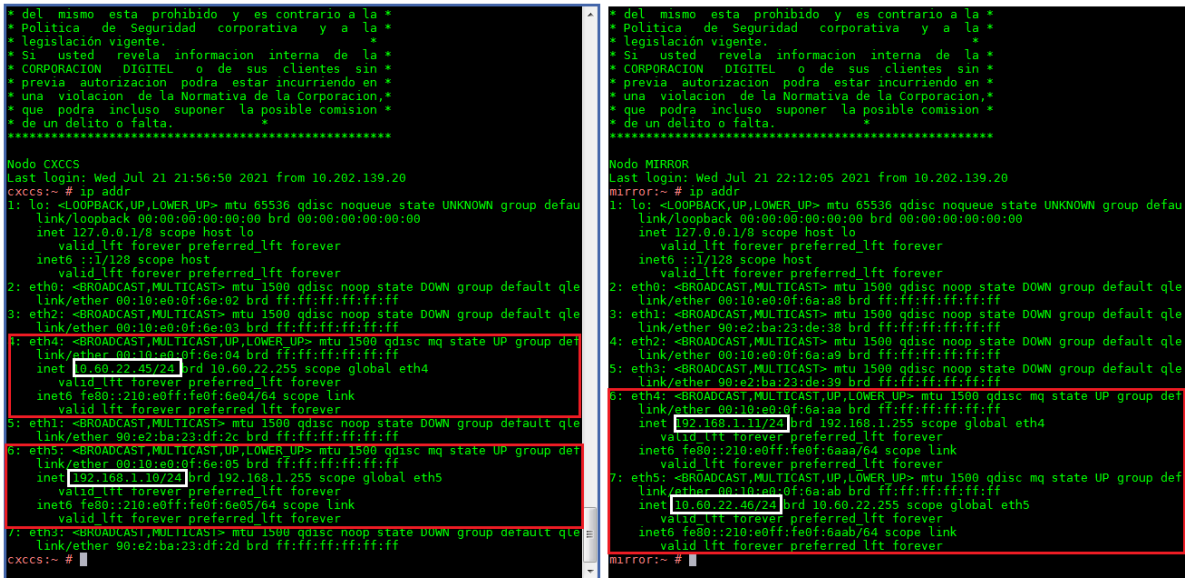


Ilustración 44. Puertos de red configurados en ambos servidores.

V.5.4 Configuración ISCSI

Este objetivo se satisface al configurar el servicio como objetivo en el servidor que cuenta con S.O. Windows server 2016 y como iniciador en los servidores de este Proyecto, a continuación, se aprecia la configuración final de ambos roles de este servicio.

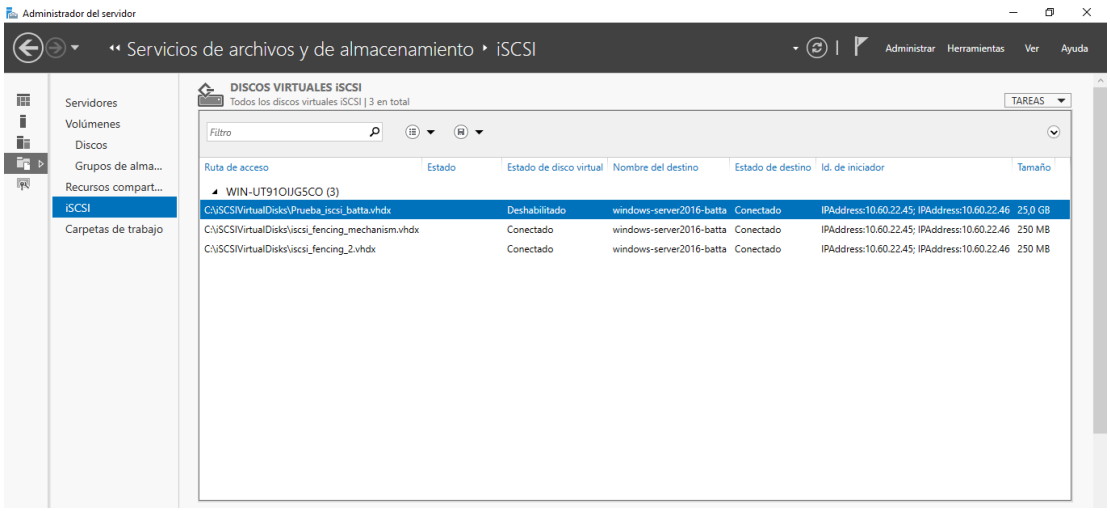


Ilustración 45. Servicio ISCSI como objetivo.

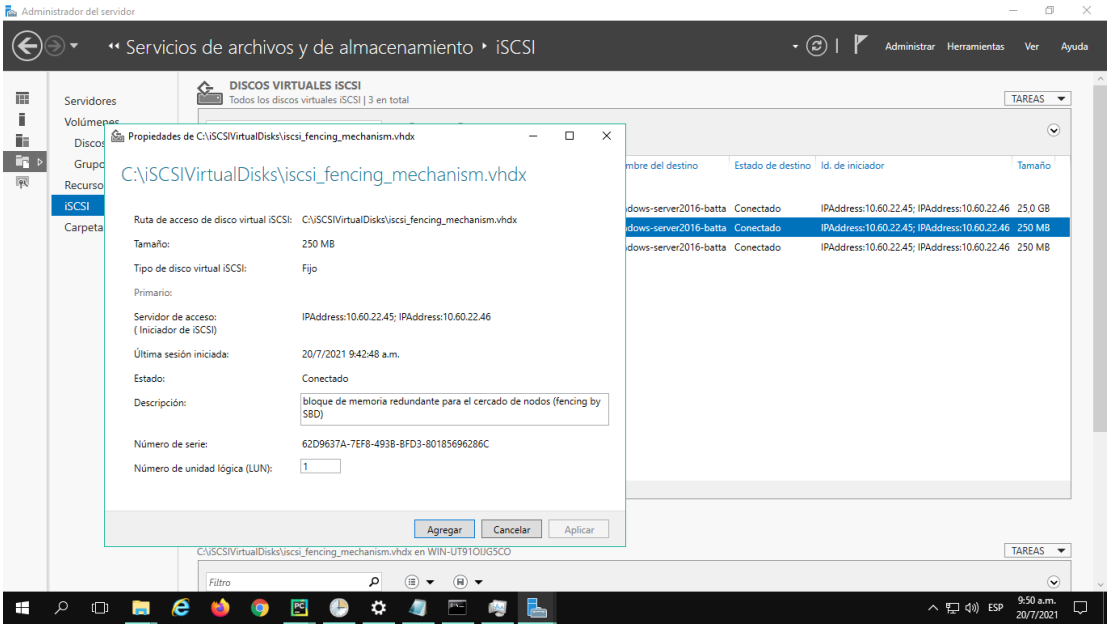


Ilustración 46. Propiedades del primer bloque de memoria provisto a través de ISCSI.

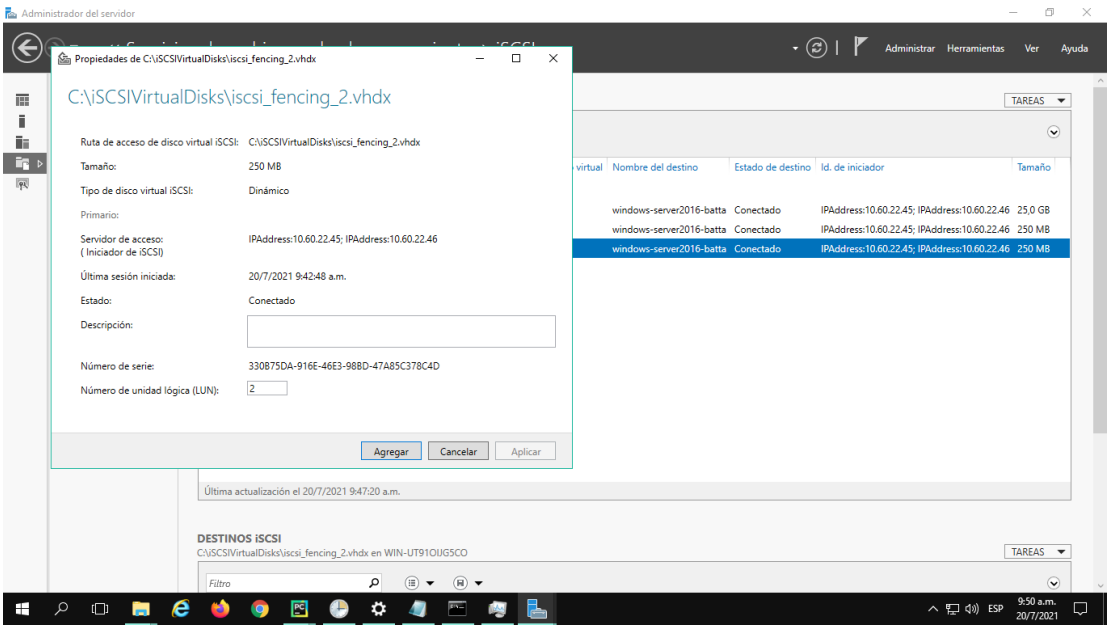


Ilustración 47. Propiedades del segundo bloque de memoria provisto a través de ISCSI.

Ilustración 48. Servicio ISCSI como iniciador en los servidores cxccs y mirror.

V.5.5 Configuración del software de Alta Disponibilidad (HA)

Este objetivo se desarrolló con éxito al concluir la configuración de las etapas necesarias hasta poner en funcionamiento el servicio de alta disponibilidad, para ello se configuró antes el SBD y el Watchdogtime.

```
SBD_DEVICE="/dev/mapper/360003ff44dc75adcbfd380185696286c;/dev/mapper/360003ff44dc75adc98bd47a85c378c4d"
SBD_OPTS=-W
```

Ilustración 49. Dispositivo SBD alimentado por los bloques de memoria provistos a través de ISCSI.

Se puede apreciar el serial de los bloques de memoria utilizados son los que aparecen en la ilustración 46 y 47 en donde se configuró ISCSI como objetivo.

```
cxccs:~ # lsmod | egrep "(wd|dog)"
iTCO_wdt               16384  0
iTCO_vendor_support    16384  1 iTCO_wdt
softdog                16384  2
cxccs:~ # lsmod | grep dog
softdog                16384  2
```

Ilustración 50. Watchdog.

Se debe recordar que SBD es el demonio que alimenta al Watchdog y este vigila constantemente los nodos en caso de que se requiera cercar a uno de ellos si ocurre alguna incidencia.

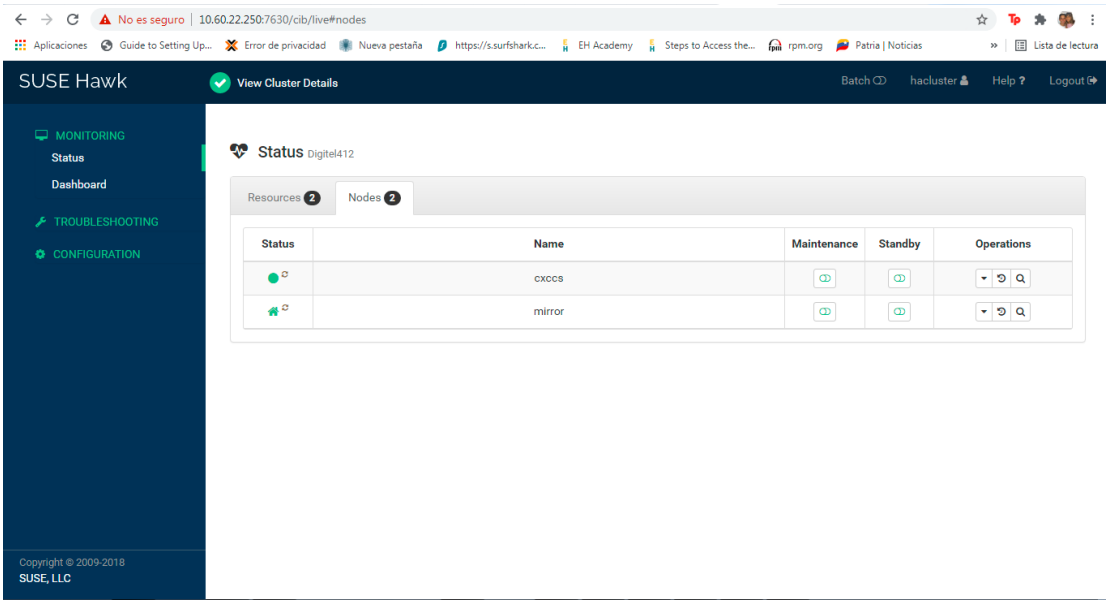


Ilustración 51. Nodos disponibles en el Servicio de Alta Disponibilidad.

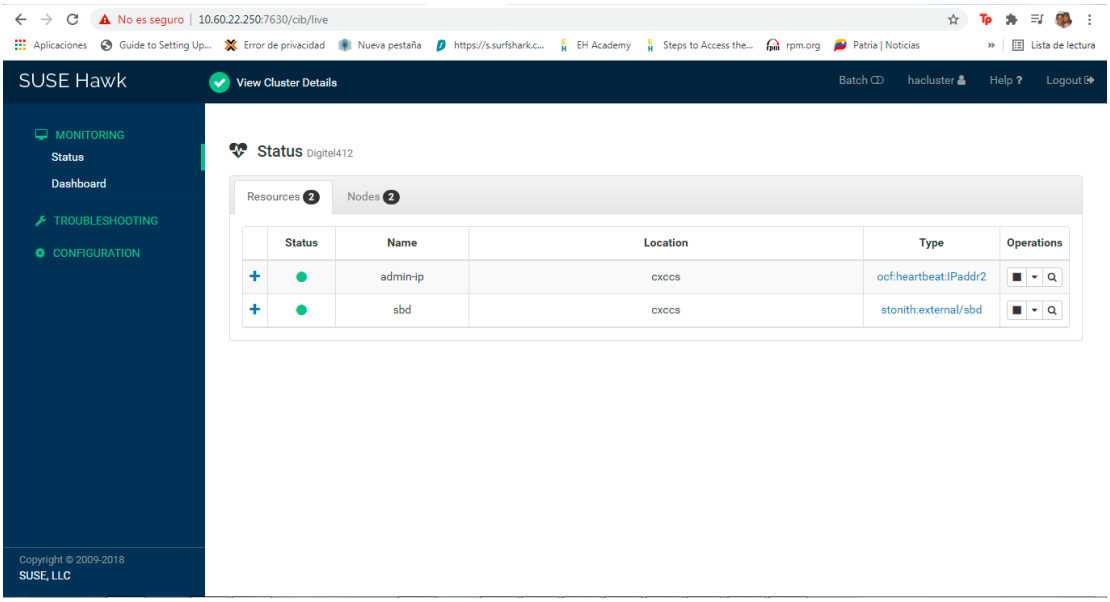


Ilustración 52. Recursos disponibles en el Servicio de Alta Disponibilidad.

En las ilustraciones 46 y 47 se aprecian los nodos y recursos disponibles bajo el servicio de Alta Disponibilidad a través de la interfaz web provista por el servicio de Hawk2.

V.5.6 Instalación y configuración física

Se cumplió con este objetivo al lograr las configuraciones físicas necesarias que desenvolvieron en la instalación de los servidores que integran el sistema implementado, partiendo desde el estudio efectuado en la sede de Sartenejas para la ubicación física tanto en la habitación como en el rack, ubicación de tomas simples y tomas con respaldo eléctrico y conexión al Switch de la Corporación el cual proveerá de acceso a la red corporativa, a continuación, se evidencia lo expuesto.

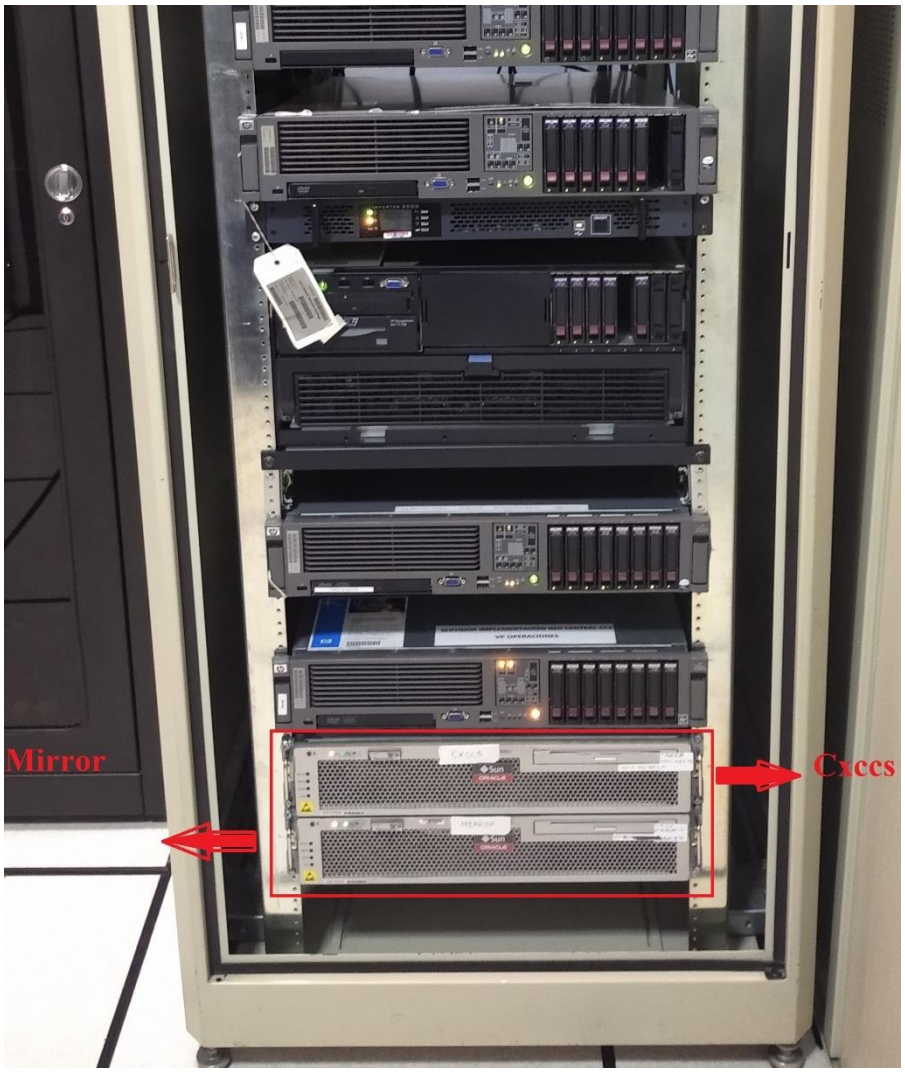


Ilustración 53. Instalación en Rack.

Ilustración 54. Conexión de puertos de red y de alimentación.

Ilustración 55. Conexión a los puertos 7 y 8 del Switch.

V.6 Pruebas de funcionamiento y operatividad del servidor.

El desarrollo de esta fase permitió la detección de fallas en el sistema, la aplicación de correcciones pertinentes, cuantificar el tiempo requerido para la ejecución de los scripts en situaciones controladas y efectuar análisis en base a los resultados obtenidos que se muestran a continuación:

V.6.1 Prueba de discos

Para corroborar de forma práctica la configuración de discos en RAID 1 se extrajo uno de los dos discos duros de uno de los servidores, en este caso el del servidor mirror.

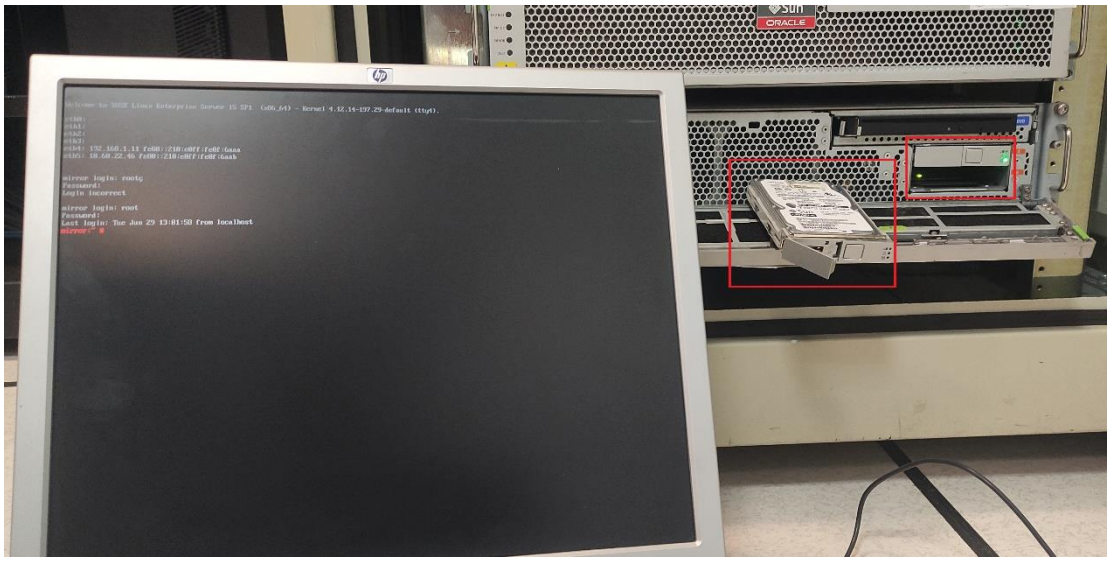


Ilustración 56. Prueba de arreglo de discos RAID 1.

V.6.2 Redundancia de puertos y resolución de Hostname

Los resultados de esta prueba se obtuvieron mediante la ejecución del comando ping entre los puertos configurados en ambos servidores.

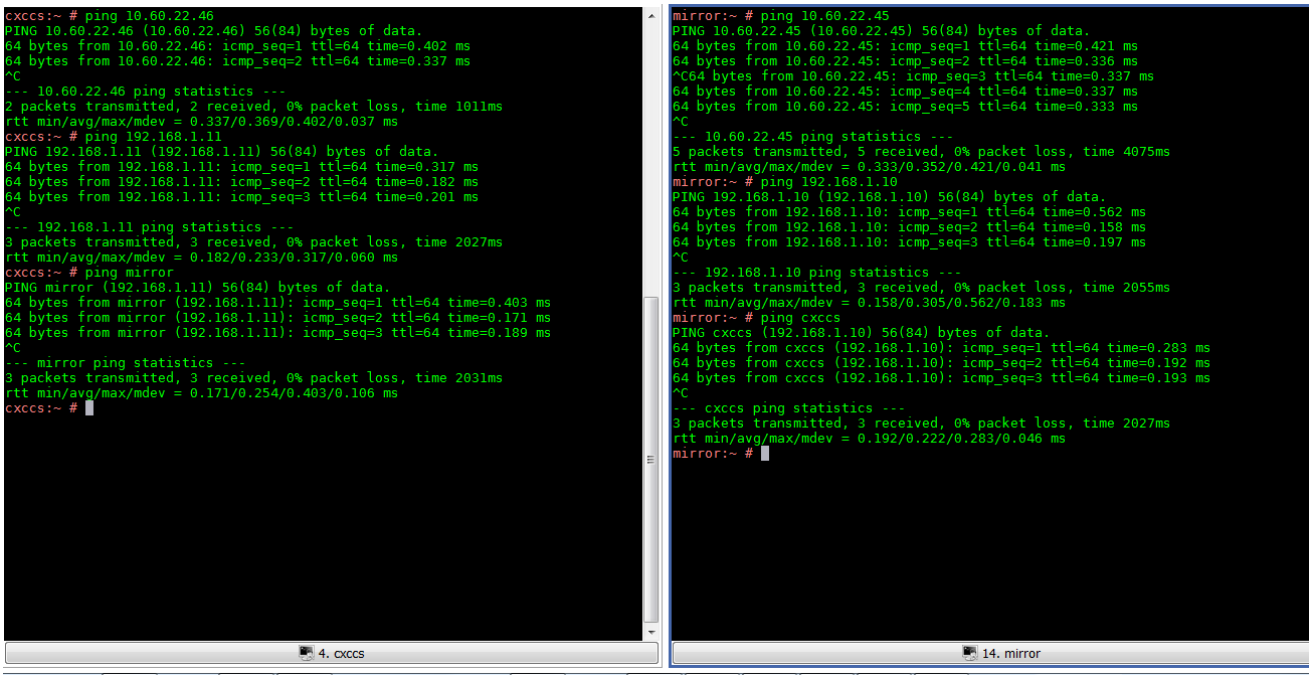


Ilustración 57. Prueba de comunicación entre los distintos puertos y de resolución de hostname.

V.6.3 Acceso remoto

Los resultados obtenidos se pueden apreciar en la ilustración 56 ya que, las pruebas se ejecutaron a través de SSH a los servidores.

V.6.4 Acceso remoto a través de la IP virtual

En el momento en el que se efectuó esta prueba, el recurso IP virtual se encontraba alojado en el nodo cxccs que es el que estaba en estado Activo, lo cual podemos constatar en la siguiente prueba.

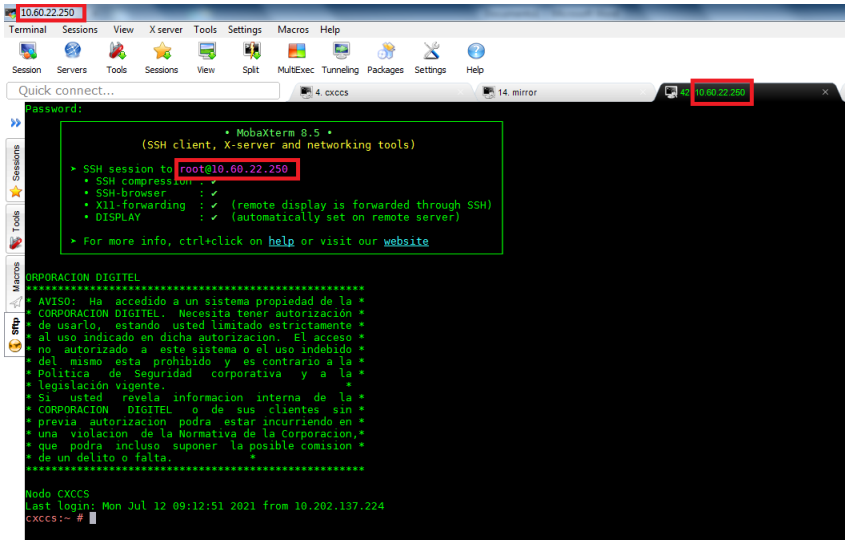


Ilustración 58. Prueba de conexión a través de SSH a la IP virtual

V.6.5 Reinicio o cercado del nodo

Como se explicó en el capítulo IV.6.6, en esta prueba se cerca el nodo en estado activo, constatando la correcta ejecución del ciclo de reinicio, ocasionándole al nodo que previamente estaba en espera pase a estar en estado activo y, además corroborar que en efecto los recursos son migrados al nuevo nodo activo. Una vez culminado el ciclo de reinicio, el nodo se reintegrará al clúster en estado pasivo. A continuación, se presentan los resultados obtenidos en esta prueba:

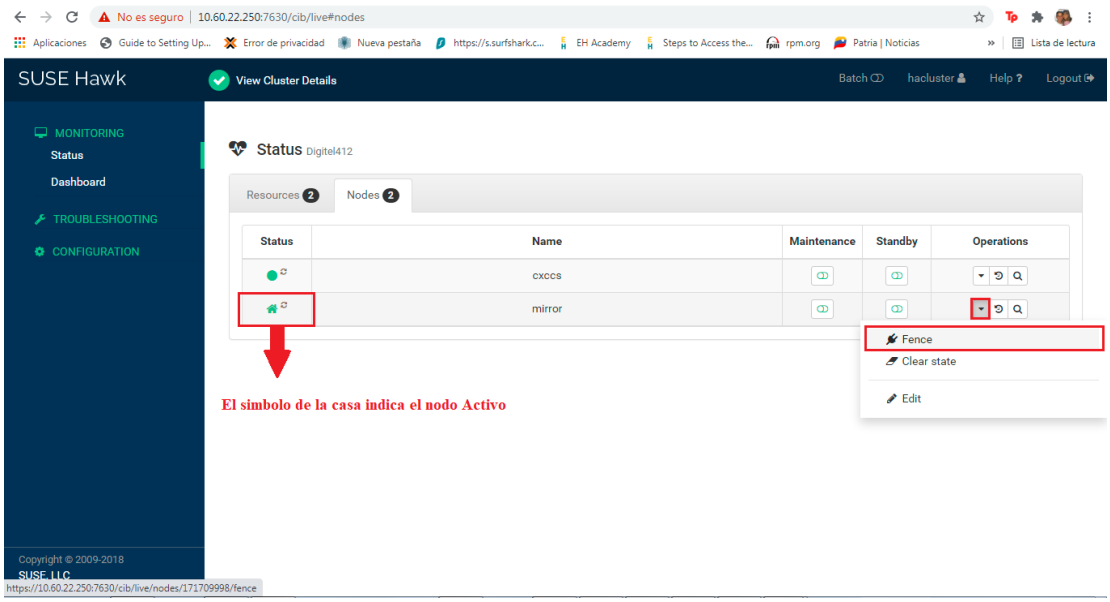


Ilustración 59. Cercado del nodo mirror.

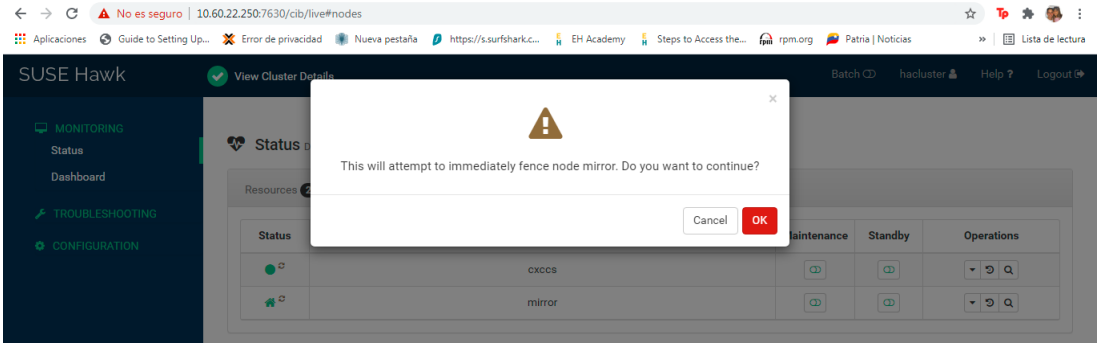


Ilustración 60. Confirmación del cercado.

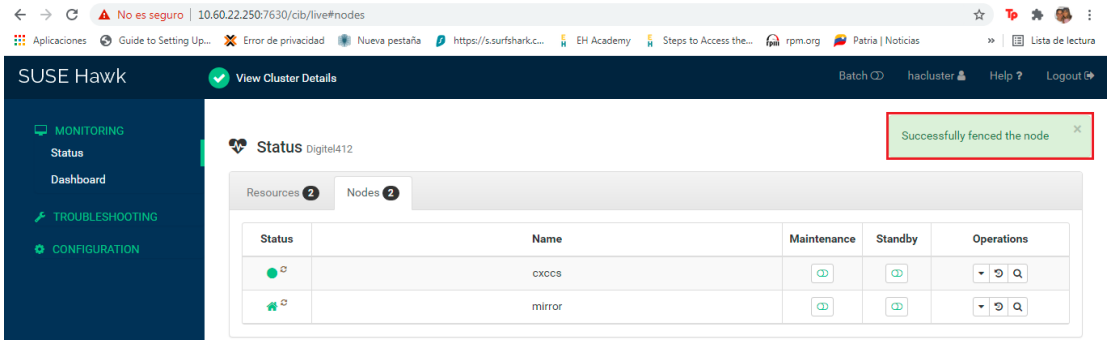


Ilustración 61. El servicio Hawk nos confirma que el nodo fue cercado.

En la ilustración 60 aparece el mensaje de confirmación alegando que el nodo fue cercado, sin embargo, al estar en la dirección IP virtual se requiere refrescar la página y reingresar las credenciales para apreciar el cambio efectuado como se muestra en la siguiente ilustración:

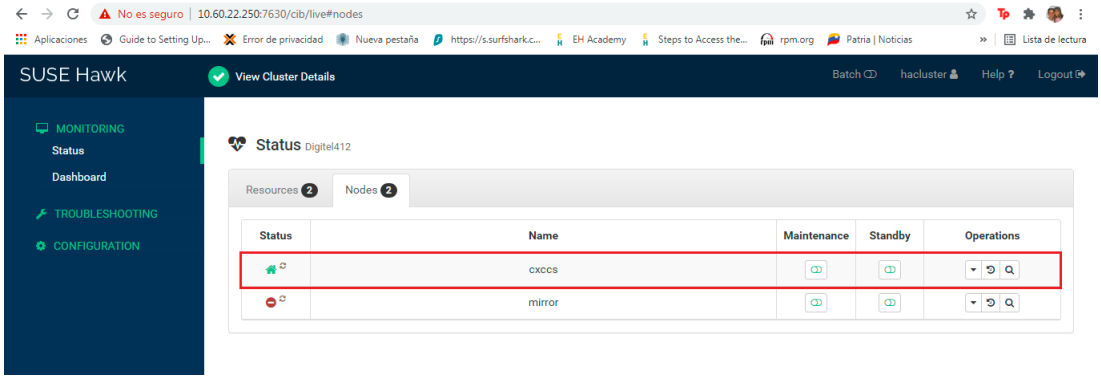


Ilustración 62. Verificación visual de que el nodo fue cercado.

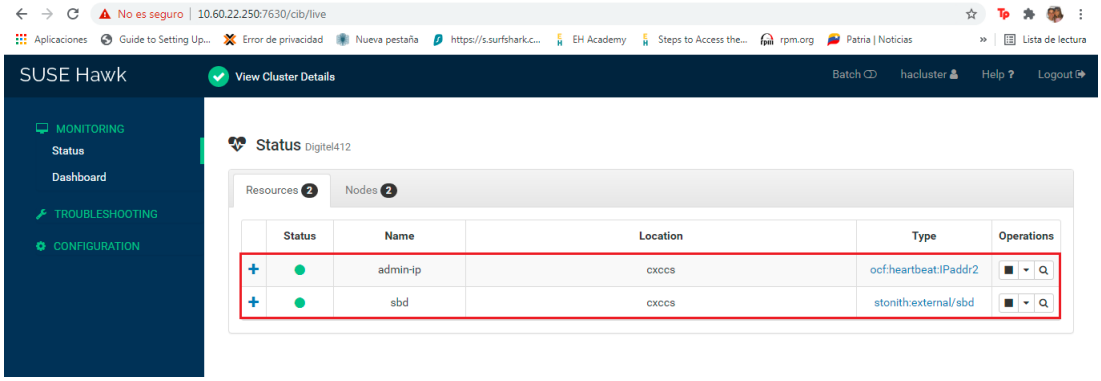


Ilustración 63. Recursos migrados con éxito al nodo cxccs.

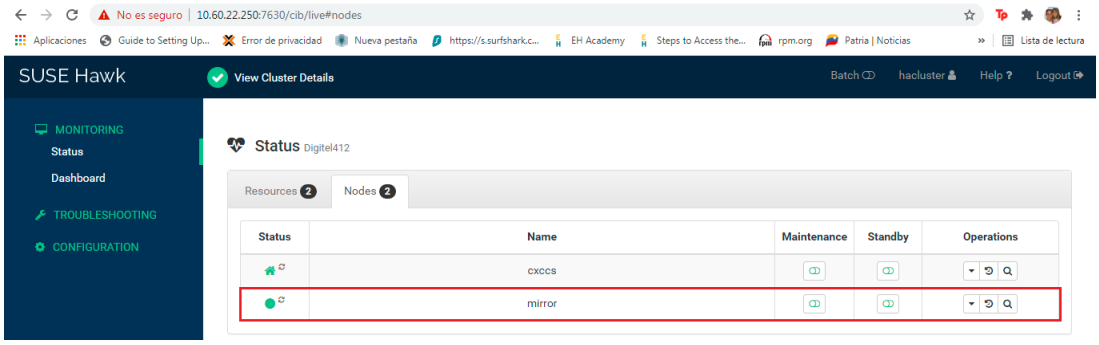


Ilustración 64. Nodo mirror en espera.

V.6.6 Ejecución de Scripts con lista de abonados

	Cantidad de abonados. resultados en minutos				
Comando ejecutado	1k	5k	10k	50K	100k
LST SUB día	0.33	1.75	3.73	17.9	33.48
LST SUB noche	0.33	1.77	3.5	30.4	58.38
LST SUB DETAIL=TRUE día	0.82	4.12	8.18	40.72	74.62
LST SUB DETAIL=TRUE noche	0.77	4.02	8.15	51.5	97.57

Tabla 7. Resultados en base a la ejecución de scripts primera ronda de día y en la noche.

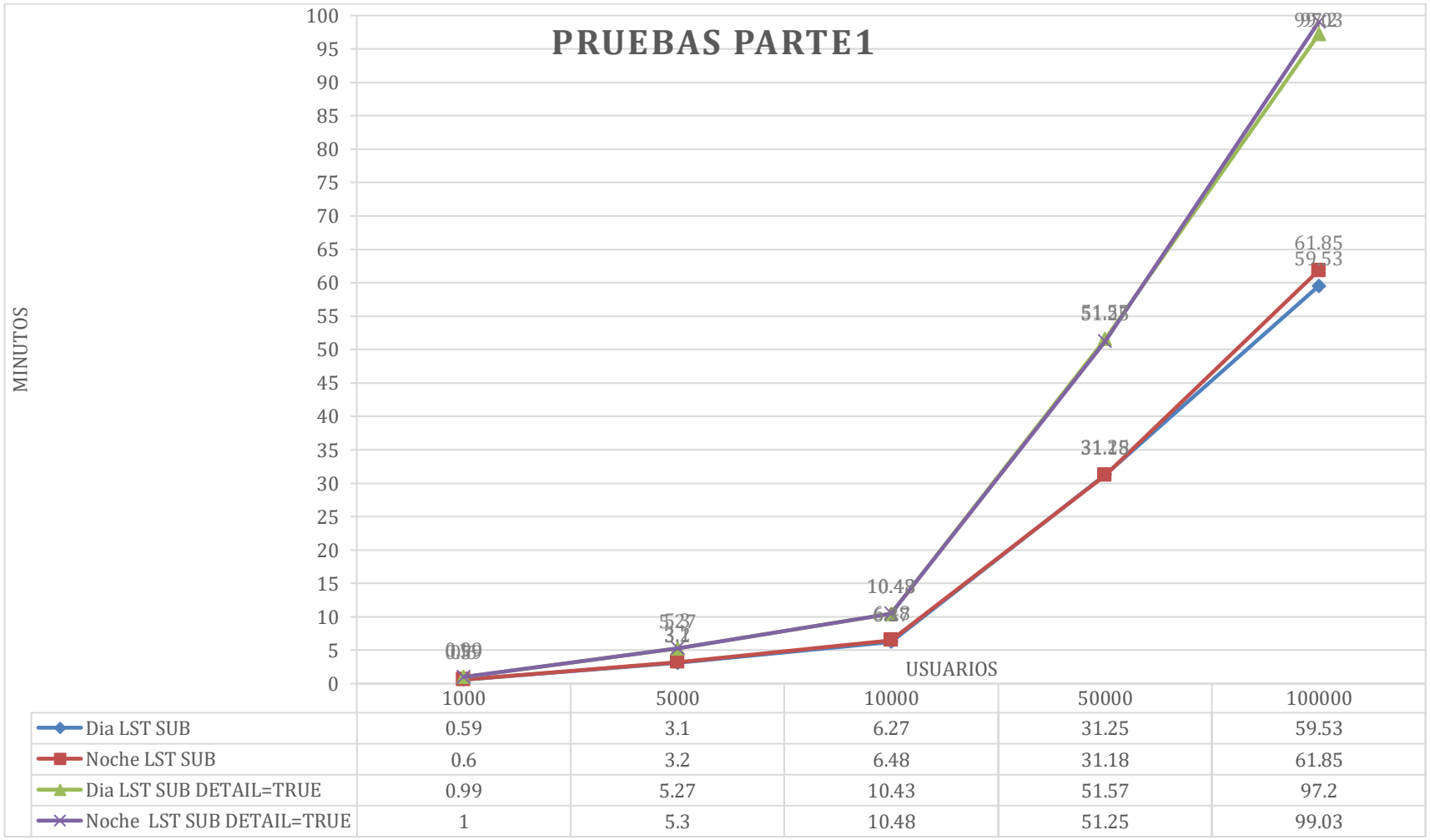
	Cantidad de abonados. resultados en minutos				
Comando ejecutado	1k	5k	10k	50K	100k
LST SUB día	0.6	3.18	6.25	31.63	58.87
LST SUB noche	0.95	3.27	6.25	30.1	58.73

LST SUB DETAIL=TRUE día	1	5.33	10.75	52.07	99.1
LST SUB DETAIL=TRUE noche	1	5.15	10.4	51.73	96.37

Tabla 8. Resultados en base a la ejecución de scripts segunda ronda de día y en la noche.

	Cantidad de abonados. resultados en minutos				
Comando ejecutado	1k	5k	10k	50K	100k
LST SUB día	0.59	3.1	6.27	31.25	59.53
LST SUB noche	0.6	3.2	6.48	31.18	61.85
LST SUB DETAIL=TRUE día	0.99	5.27	10.43	51.57	97.2
LST SUB DETAIL=TRUE noche	1	5.3	10.48	51.25	99.03

Tabla 9. Resultados en base a la ejecución de scripts tercer ronda de día y en la noche.



	1000	5000	10000	50000	100000
Dia LST SUB	0.59	3.1	6.27	31.25	59.53
Noche LST SUB	0.6	3.2	6.48	31.18	61.85
Dia LST SUB DETAIL=TRUE	0.99	5.27	10.43	51.57	97.2
Noche LST SUB DETAIL=TRUE	1	5.3	10.48	51.25	99.03

Ilustración 65. Resultados de ejecución de la primera ronda de pruebas.

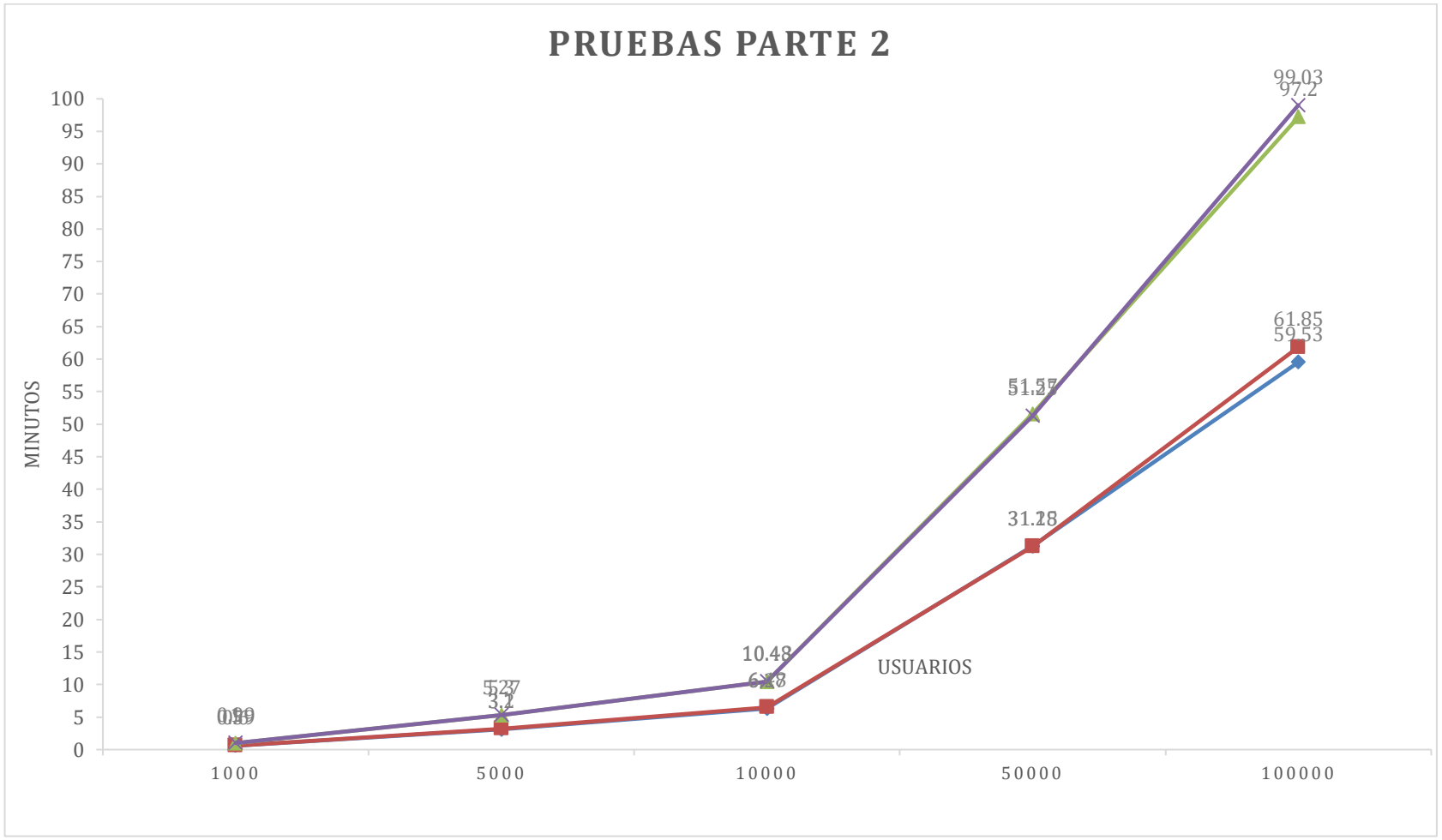


Ilustración 66. Resultados de ejecución de la segunda ronda de pruebas.

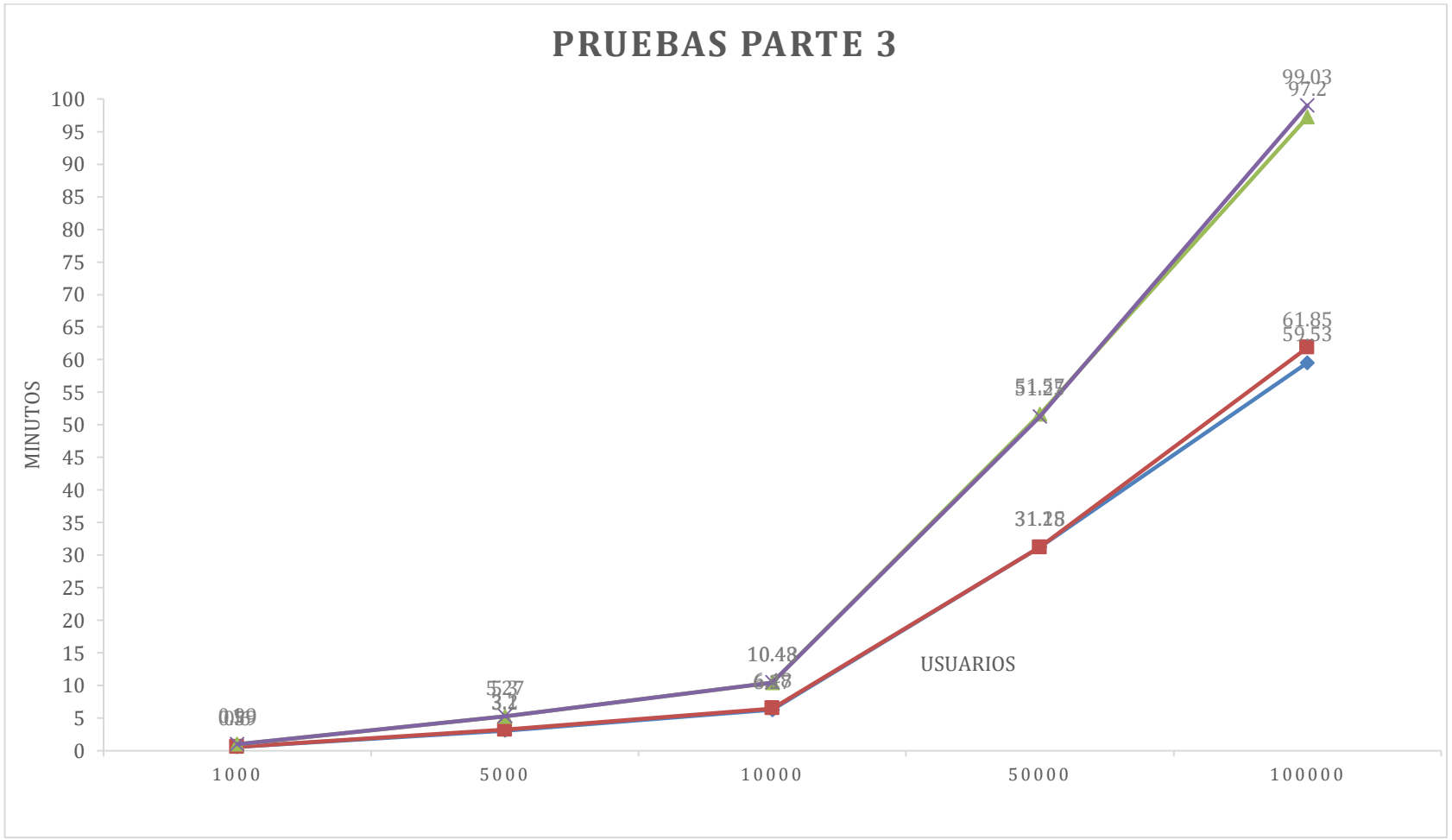


Ilustración 67.Resultados de ejecución de la tercera ronda de pruebas.

A partir de los resultados obtenidos se puede analizar lo siguiente:

1. Las diferencias de tiempo pueden variar independientemente de la hora en la que se ejecuten los scripts, sin embargo, es recomendable hacer las ejecuciones en ventanas de tiempo nocturnos debido a que en principio el Single SDB tiene menor tráfico, aunque esto puede variar acorde a la interacción de distintos departamentos con este equipo.
2. Algunos resultados obtenidos en la primera sesión de pruebas durante el día, se puede constatar la reducción en el tiempo de ejecución de los comandos, esto se debe a que en esa ventana de tiempo el Single SDB presentaba poco tráfico de comandos, es decir, pocas personas estaban interactuando con este equipo.

Capítulo VI

Conclusiones y Recomendaciones

En el presente capítulo se exponen las conclusiones obtenidas al finalizar el presente Trabajo Especial de Grado, junto con las recomendaciones que se consideran necesarias para el uso de este proyecto y para el desarrollo de futuros proyectos similares.

VI.1 Conclusiones

Las investigaciones realizadas permitieron una correcta toma de decisiones al implementar el sistema, desempeñando los requerimientos mínimos de operación para cumplir con los objetivos planteados. De esta forma, para efectuar una correcta implementación, se deben adquirir conocimientos básicos sobre los tópicos relacionados a redes, protocolos de red, servidores, softwares de alta disponibilidad, arreglo de discos y programación, debido a que, con la correcta toma de decisiones, es posible adquirir un sistema capaz de consultar el registro de usuarios de la Corporación Digitel a la vez que se ofrece una alta calidad del servicio.

Los estudios llevados a cabo en conjunto con el departamento de Operación y mantenimiento (O&M) Red Central, permitieron la modificación de los scripts para que el operador del departamento tenga la libertad de introducir por teclado cualquier comando ejecutable del Single SDB que se desee ejecutar, brindando un mayor alcance y la optimización de este Proyecto Especial de Grado. A su vez se puede concluir en que los lenguajes de programación son una poderosa herramienta que brinda una diversidad de soluciones en esta era de la tecnología.

Es importante resaltar que, al diseñar e implementar este Trabajo Especial de Grado se contó con el apoyo de dos coordinaciones de la Vicepresidencia de Operaciones y una coordinación de la vicepresidencia de Sistemas de la Corporación Digitel. Gracias a esta colaboración se apreció el alcance que tiene este proyecto por proveer un sistema capaz de efectuar consultas en el registro de usuarios y que a su vez cuente con una solución de Alta Disponibilidad, ofreciendo así un sistema estable, fiable y capaz de superar fallas de conexión, perdida de información, energía y fallas de cerebro dividido en los nodos.

El montaje y configuración de los servidores se llevo a cabo a través de las configuraciones iniciales, partiendo en la configuración de discos a través de un arreglo RAID 1, lo que permite superar fallas que puedan presentarse en uno de los discos; La instalación del sistema operativo y el software de

Alta Disponibilidad, en donde se procede a configurar los demás requerimientos, protocolos y servicios, también provee la CLI a través de la cual se ejecutan los scripts de programación; Configuración de ISCSI como iniciador y objetivo, suministrando la alimentación del demonio SBD utilizado para el monitoreo y cercado de los nodos; Configuración del software de Alta Disponibilidad que garantiza fiabilidad en el sistema al encarar fallas críticas en los nodos mediante el uso de STONITH y SBD, forzando al nodo afectado a iniciar el ciclo de reinicio y migrar los recursos. El producto del desarrollo de estos pasos brinda el sistema planteado en los objetivos a modo de prueba, debido a que, se requiere ejecutar la instalación física y llevar a cabo pruebas para dejar en producción el sistema.

El estudio efectuado en la sede de Sartenejas permitió corroborar la disponibilidad en los racks de servidores, disposición de puertos de conexión al switch, tomas de energía comunes y con respaldo, obteniendo la información necesaria para ejecutar la instalación del sistema desarrollado a lo largo de este Trabajo Especial de Grado.

Las pruebas de funcionamiento y operatividad, como en todo sistema, son un requerimiento fundamental que permite contemplar el comportamiento, tiempos de reacción o ejecución y corroborar que los cambios o configuraciones perpetradas hayan surtido efecto, repitiendo una y otra vez hasta lograr los objetivos planteados. A lo largo de este objetivo pude llegar a las siguientes conclusiones:

1. La configuración matricial de discos RAID 1 es uno de los arreglos de mayor relevancia en el mundo de los servidores, ya que, tener respaldos previendo la falla repentina de un disco es indispensable.
2. Un sistema que preste cualquier tipo de servicios debe contar con al menos dos (2) puertos de red configurados, debido a que, al igual que un disco, una tarjeta de red también puede fallar.
3. La interacción remota con todo sistema es vital y cada vez es mayor la demanda de servicios a distancia.
4. El software de alta disponibilidad es una herramienta avanzada y sin igual, usada casi en su totalidad en escenarios empresariales permitiendo la replicación de la información, respaldo del sistema en su totalidad al tener nodos activos/pasivos, autogestión en caso de que uno de los nodos falle, reiniciando el nodo fallido y migrando los recursos de un nodo a otro dentro del clúster de ser necesario.

Cabe destacar que las pruebas llevadas a cabo fueron exitosas, concluyendo así, con un sistema en producción que le agrega valor a la Corporación Digitel, permitiendo escalabilidad y expansión de servicios si así se desea.

VI.2 Recomendaciones

Si se desea aumentar la velocidad de transmisión de la información, se recomienda instalar una o más tarjetas de red que utilicen como medio de transmisión la fibra óptica.

Si el departamento de Operación y Mantenimiento (O&M) es el único interactuando con el Single SDB se puede garantizar tiempos menores de ejecución.

Si se desea aumentar el alcance de este proyecto, es recomendable la implementación de una página web a través de la cual se pueda ejecutar los scripts de programación sin la necesidad de una terminal.

Referencias Bibliográficas

[1] Z. M. P. a. P. I. Becvar, Redes móviles, Czech Republic, pp. 23-74.

[2] Corporacion Digitel, «Elementos de la red central,» Caracas, 2015.

[3] SingleSDB Product Documentation, «Hedex,» Huawei, 02 06 2012. [En línea]. Available: http://10.60.22.91:52199/hedex/hdx.do?lib=31185441&v=02&tocLib=31185441&tocV=02&id=cn_22_18_000011&tocURL=resources%252fbe%252fmaintenance%252fsecurity%255fmgt%252fcn%255f22%255f18%255f000011%252ehtml&p=t&fe=1&ui=3&keyword=pgw. [Último acceso: 09 09 2019].

[4] M. Gallo y W. Hancock, Comunicación entre Computadoras y Tecnología de Redes, Mexico: Ediciones Paraninfo, 2002.

[5] A. a. W. D. Tanenbaum, Redes de computadoras, Naucalpan de Juárez: Pearson Educación, 2012.

[6] D. Mills, Computer Network Time Synchronization, Boca Raton : CRC Press, 2006.

[7] S. LLC, «SUSE,» SUSE, 01 12 2019. [En línea]. Available: https://documentation.suse.com/sles/15-SP1/html/SLES-all/book-storage.html. [Último acceso: 10 12 2020].

[8] R. S. Daniel J. Barrett, SSH, The Secure Shell: The Definitive Guide, O'Reilly, 2001.

[9] E. A. Marchionni, Administrador de servidores, Buenos Aires : Fox Andina, 2011.

[10] P. Christensson, «Tech Terms,» 17 08 2017. [En línea]. Available: www.techterms.com/definition/mirror. [Último acceso: 15 12 2019].

[11] An Oracle White Paper, «Oracle's Sun Netra X4250 and X4270 Server,» 2010.

[12] Copyright © 2006–2016 SUSE LLC and contributors., «SUSE,» 14 marzo 2016. [En línea]. Available: https://www.suse.com/documentation/sles11/singlehtml/book_sle_admin/book_sle_admin.html#sec.trouble.data. [Último acceso: 20 junio 2019].

[13] Red Hat, Inc, «2003 por Red Hat, Inc,» Red Hat, Inc, [En línea]. Available: http://web.mit.edu/rhel-doc/3/rhel-sag-es-3/s1-raid-levels.html. [Último acceso: 15 05 2019].

[14] ©2002DellComputerCorporation, «WebBIOS,» 2002.

[15] S. V. Vugt, Pro Linux High Availability Clustering, Berkeley: Apress, CA, 2014.

[16] SUSE LLC, «SUSE,» 09 12 2019. [En línea]. Available: https://documentation.suse.com/sle-ha/15-SP1/html/SLE-HA-all/book-sleha-guide.html. [Último acceso: 15 12 2019].

[17] Red Hat , Inc., «Red Hat,» 2002. [En línea]. Available: https://access.redhat.com/documentation/es-es/red_hat_enterprise_linux/6/html/configuring_the_red_hat_high_availability_add-on_with_pacemaker/ch-fencing-haar. [Último acceso: 1 12 2019].

[18] B. D. F. L. W. Tom Christiansen, «Programming Perl,» O´REILLY , Sebastopol, 2012.

Anexos

Por motivos prácticos, los scripts usados en este proyecto se encuentran en el siguiente link de Google drive: https://drive.google.com/drive/folders/1vPaDYmwMHghqwjUvPYmRzsgP-uesDHLq?usp=sharing