



BENEMÉRITA UNIVERSIDAD AUTÓNOMA DE PUEBLA

**FACULTAD DE CIENCIAS DE LA
COMPUTACIÓN**

**Mejora y Actualización de la
Infraestructura de una Red
Centralizada en GRID para Incrementar
la Eficiencia en el Procesamiento
Distribuido**

**TESIS PROFESIONAL
para obtener el título de
LICENCIADO EN INGENIERÍA EN
INGENIERÍA EN CIENCIAS DE LA
COMPUTACIÓN**

**PRESENTA
Raymundo Martínez Marcos**

**Director de tesis:
Dr. Mariano Larios Gómez**

Julio 2024

Dedicatoria:

A mis padres María Modesta Félix de Marcos Alducin y José Raymundo Fernando Martínez Martínez a mis hermanos y hermanas Miriam Martínez Marcos, José Emmanuel Martínez Marcos y Laura Martínez Marcos, que siempre han estado presentes en mis momentos de necesidad y angustia, brindando todo el apoyo y cariño que un hijo puede pedir, a ellos por enseñarme que incluso en los momentos más difíciles de la vida debes levantarte y seguir peleando, porque la vida es una pelea constante, por fungir como el faro y la luna que me guiaron en el camino de oscuridad, por ser las bases que me ayudaron a llegar hasta este momento, en el cual dejo una etapa muy importante de mi vida atrás alcanzando una meta más, pero abro las puertas de un futuro incierto y desconocido pero que con sus enseñanzas estoy seguro lograré afrontar con el rostro en alto.

El presente trabajo está dedicado a toda mi familia, a mis padres, a mi hermana, profesores y amigos que me inspiraron tanto en mi vida diaria como en la estudiantil a redactar este trabajo, ellos son quienes me dieron grandes enseñanzas y experiencias, ciertamente convirtiéndose así en los verdaderos protagonistas de este “sueño alcanzado”.

Agradecimientos:

Me van a faltar páginas para agradecer a las personas que se han involucrado en la realización de este trabajo, sin embargo merecen agradecimiento de todo corazón mis padres, familia y amigos, de igual manera mis agradecimientos a la Benemérita Universidad Autónoma de Puebla y a toda la Facultad de Ciencias de la Computación, a mis profesores, en especial al profesor Mariano Larios Gómez por su paciencia dedicación, motivación, criterio y amistad, el cual es el principal colaborador durante todo este proceso, quien con su dirección, conocimiento, enseñanza y colaboración permitió el desarrollo de este trabajo, ha hecho fácil esta travesía y un privilegio el trabajar a su lado.

Mi profundo agradecimiento a todas las autoridades y personal que conforman a la Facultad de Ciencias de la Computación.

También deseo aprovechar este momento para agradecer a Dios, por haberme dado la vida y permitirme el haber llegado hasta este momento tan importante de mi formación profesional.

Por último, agradezco al Laboratorio de Desarrollo Tecnológico y al Laboratorio Nacional de Supercómputo del Sureste de México que pertenece a la red de laboratorios nacionales CONACyT, por los recursos computacionales, el apoyo y la asistencia técnica.

“A Todos Muchas Gracias”

Índice general

Dedicatoria:	2
Agradecimientos:	3
Índice general	4
1 Introducción	1
1.1 Resumen.	1
1.2 Antecedentes del Proyecto.	5
1.3 Objetivos.	6
1.4 Metodología.	8
1.6 Infraestructura.	11
2 Estado del arte.	15
3 Configuración de una red GRID de Oracle rac 2cr1 (12.1.0.2) en solaris sparc-64	18
3.1 Configuración automática de Oracle Solaris con Oracle Database.	33
3.2 Comprobación de la configuración de los paquetes de requisitos previos de Oracle Database	34
3.3 Topología	38
3.4 Componentes clave de la infraestructura Grid de Oracle	6
3.5 Beneficios de usar la infraestructura de Grid	9
4 Resultados	25
4.1 Parches, Drivers y Dns	26
4.2 Minimizar el tiempo de inactividad	27

4.3	Sertificados SSL	33
4.4	Sincronización (sistemas paralelos)	38
	Identificación del tipo de deterioro de datos	40
4.5	Reparación de daños en las agrupaciones de almacenamiento de ZFS	42
4.6	RECOMENDACIONES GENERALES:	43
	A NIVEL DE BASE DE DATOS:	43
	A NIVEL DE TABLAS:	43
	A NIVEL DE CAMPOS:	45
	A nivel de tipos de datos para sql server:	45
4.7	Interfaz uniforme	46
4.8	Resultados	47
5	Conclusiones.	58
	Bibliografía.	62

1 Introducción

1.1 Resumen.

Este proyecto de investigación de tesis se enfoca en abordar de manera integral el desafío de mejorar la infraestructura y modernización de redes tipo GRID (grid computing), con el objetivo de potenciar su eficiencia y capacidad para el procesamiento distribuido de datos y cálculos intensivos. Se reconoció la necesidad imperante de optimizar y revitalizar las redes GRID existentes, en respuesta al crecimiento exponencial en la demanda de recursos computacionales.

Para lograr este propósito, se diseñó y ejecutó un extenso plan de trabajo que abarcó múltiples aspectos relacionados con la infraestructura y configuración de las redes GRID. En primer lugar, se llevó a cabo un exhaustivo análisis del hardware del servidor de instalación de Oracle Grid Infrastructure, con el fin de verificar su idoneidad y compatibilidad con los requerimientos del sistema. Este proceso implicó la evaluación detallada de componentes clave, como la capacidad de almacenamiento, la memoria y otros recursos esenciales para el funcionamiento óptimo de la infraestructura GRID.

Además, se dedicó una atención especial a la configuración del entorno para Oracle Grid Infrastructure y Oracle RAC (Real Application Clusters). Esto incluyó la realización de tareas de configuración de red, almacenamiento y protocolos de tiempo de red para garantizar la sincronización adecuada entre los nodos del clúster. Asimismo, se llevaron a cabo acciones preparatorias en los

nodos del clúster, como la creación de cuentas de usuario y la preparación del almacenamiento compartido, para asegurar una implementación sin contratiempos.

Otro aspecto fundamental del proyecto fue la instalación de la infraestructura de Oracle Grid, que implicó seguir un método específico y cumplir con los requisitos del sistema operativo. Durante este proceso, se prestaron especial atención a la ejecución de ASMCA para la creación de grupos de discos necesarios para el funcionamiento de Oracle RAC, así como a la instalación del software RDBMS para completar la configuración del entorno GRID.

Este proyecto representó un esfuerzo exhaustivo y meticuloso para mejorar la infraestructura y modernización de redes GRID, con el fin de maximizar su eficiencia y capacidad de procesamiento distribuido. Los resultados obtenidos proporcionaron una sólida base para futuras investigaciones y desarrollos en el campo de la computación distribuida, contribuyendo así al avance continuo de esta área tecnológica clave.

En el mundo empresarial contemporáneo, la gestión eficiente y segura de la información es fundamental para el éxito organizacional. Oracle ha emergido como un líder indiscutible en la oferta de soluciones tecnológicas avanzadas, destacando especialmente en el ámbito de infraestructura de red. Este informe tiene como objetivo proporcionar una visión detallada y analítica de la infraestructura de red Oracle, explorando sus componentes clave, su arquitectura, y su capacidad para impulsar operaciones empresariales eficientes y seguras.

Las organizaciones modernas buscan optimizar sus procesos y potenciar la toma de decisiones basada en datos, por lo que la infraestructura de red Oracle se erige como una piedra angular esencial. Este reporte se sumerge en los elementos esenciales que conforman esta infraestructura, examinando sus características, ventajas y consideraciones clave para garantizar un desempeño óptimo. Desde la gestión de bases de datos hasta la seguridad de la red, el informe proporciona una comprensión integral de cómo Oracle ha perfeccionado su infraestructura de red para satisfacer las demandas de un entorno empresarial en constante evolución [1].

En el ámbito de las redes de computadoras, se examina cómo se conectan los dispositivos entre sí y la importancia de las direcciones IPv4 para la identificación y comunicación dentro de estas redes. Se distingue entre direcciones IP privadas y públicas, fundamentales para la gestión tanto interna como externa de la red, y se analiza el concepto de IP Virtual (VIP) para una gestión más flexible y eficiente de las direcciones IP. La seguridad en la administración de redes y sistemas es otro pilar importante del informe, incluyendo medidas como la actualización constante del sistema operativo, respaldos de bases de datos (SQL), la aplicación de parches, y el uso de SSL para asegurar las comunicaciones en línea [2].

Como se menciona en [3], en cuanto al hardware y arquitecturas, se analizan las arquitecturas x86 en el contexto de Windows Server y la sincronización en sistemas paralelos para maximizar la eficiencia y rendimiento en entornos distribuidos. Los scripts de reparación son presentados como herramientas valiosas para la automatización y resolución de problemas en sistemas operativos, mientras que Oracle Solaris se destaca como un sistema operativo

robusto y confiable para entornos empresariales que requieren alta disponibilidad y rendimiento.

El informe también cubre las mejores prácticas para administrar bases de datos Oracle y el uso de Oracle RAC (Real Application Clusters), que permite la escalabilidad y disponibilidad de la base de datos mediante múltiples servidores. Además, se destaca el papel de las API REST en la integración y comunicación entre aplicaciones en la era de la nube y los servicios web.

Se exploran aspectos fundamentales para la operatividad segura y eficiente de una organización, como la configuración de la infraestructura de red y la implementación de Oracle Grid/Base de Datos. La configuración de usuarios, grupos y entornos para la infraestructura de red de Oracle y la base de datos es crucial, implicando la asignación cuidadosa de privilegios y medidas de seguridad. Además, se detalla la implementación de redes que conectan diferentes edificios y la gestión del tráfico mediante listas de acceso.

En el contexto de Oracle Grid Infrastructure, el informe profundiza en componentes clave como Automatic Storage Management (ASM), Clusterware y la base de datos Oracle, destacando su capacidad para garantizar alta disponibilidad, escalabilidad y gestión simplificada de recursos. Se describe el proceso de instalación de Oracle Grid Infrastructure, incluyendo la creación de clústeres y la administración de la base de datos Oracle RAC, utilizando herramientas como SRVCTL para la gestión y políticas de conmutación y recuperación ante fallos.

El proceso de desinstalación del software de Oracle, ofreciendo una guía detallada sobre las opciones y pasos para llevar a cabo esta tarea, asegurando la migración y reconfiguración adecuadas en entornos de clúster.

1.2 Antecedentes del Proyecto.

Esta tesis surgió de un proyecto admitido en el laboratorio de supercómputo LNS-BUAP, con número 201801076C. Gracias a este proyecto, que fue aceptado y terminados satisfactoriamente se obtuvieron los resultados deseados [18-21].

Proyectos que se someterán en el LNS son:

- Plataforma para la aplicación de algoritmos de planificación de procesos en tiempo real.
- Planificación y comunicación en un sistema distribuido simulando sobre una red P2P.
- SMART-SOFTWARE para extraer recursos informáticos de repositorios distribuidos

Así también, el presente trabajo de investigación se enmarca en un contexto donde la eficiencia y capacidad de procesamiento de las redes tipo GRID (grid computing) han adquirido una relevancia creciente en diversos ámbitos, como la investigación científica, la industria y la educación. Ante el incremento exponencial en la demanda de recursos computacionales para el procesamiento distribuido de datos y cálculos intensivos, se ha identificado la necesidad imperante de optimizar y revitalizar las infraestructuras existentes.

Con el objetivo de abordar esta problemática y contribuir al avance en el campo de la computación distribuida, se propone este proyecto de investigación. Este proyecto se inspira en la necesidad de mejorar la infraestructura y modernización de las redes GRID, reconociendo la importancia de maximizar su eficiencia y capacidad para satisfacer las demandas actuales y futuras de procesamiento.

Tomando en consideración este contexto y los desafíos presentes en la gestión de redes GRID, se plantea la realización de un estudio detallado sobre la infraestructura y configuración de estas redes. Este estudio se fundamenta en la idea de que la optimización y modernización de las redes GRID son aspectos fundamentales para su adecuado funcionamiento y su capacidad de adaptación a las necesidades cambiantes de la computación distribuida.

A través de una metodología rigurosa y minuciosa, se espera identificar las áreas de mejora en la infraestructura de las redes GRID y proponer soluciones innovadoras que permitan superar las limitaciones existentes. Se busca así sentar las bases para futuras investigaciones y desarrollos en el campo de la computación distribuida, contribuyendo de esta manera al avance tecnológico y científico en esta área de conocimiento.

1.3 Objetivos.

Objetivo General: Optimizar y modernizar una red tipo GRID con el fin de potenciar su eficiencia y capacidad de procesamiento centralizado, respondiendo así a las demandas actuales y futuras de computación distribuida.

En los objetivos específicos tenemos:

- Realizar un análisis exhaustivo del estado actual de la red GRID, examinando su infraestructura, arquitectura y configuraciones existentes para identificar y comprender a fondo las limitaciones que afectan su rendimiento y eficiencia operativa.

- Identificar y priorizar áreas específicas de mejora en la infraestructura de la red GRID, centrándose en aspectos clave como la escalabilidad, la seguridad, la disponibilidad y la tolerancia a fallos, con el objetivo de diseñar soluciones que aborden de manera efectiva los desafíos identificados.
- Diseñar e implementar mejoras técnicas en la infraestructura de la red GRID, tales como la optimización de la topología de red, la actualización de hardware y software, la implementación de medidas de seguridad avanzadas y la configuración de políticas de gestión de recursos, con el fin de mejorar su capacidad de procesamiento y su eficiencia operativa.
- Realizar pruebas exhaustivas de las mejoras implementadas en la red GRID, utilizando casos de prueba realistas y escenarios de carga para evaluar su rendimiento, estabilidad y capacidad de recuperación ante situaciones adversas, con el propósito de garantizar su fiabilidad y robustez en entornos de producción.
- Evaluar el impacto socioeconómico de las mejoras realizadas en la red GRID, analizando indicadores clave como la reducción de tiempos de procesamiento, la optimización del uso de recursos, la mejora en la productividad del personal y el retorno de la inversión, con el objetivo de demostrar el valor agregado de las mejoras implementadas.
- Documentar detalladamente todas las actividades realizadas durante el proceso de mejora de la red GRID, incluyendo los análisis realizados, las soluciones propuestas, las implementaciones realizadas y los resultados obtenidos, con el fin de proporcionar un recurso completo y accesible para futuras referencias y auditorías.

1.4 Metodología.

El uso de las nuevas tecnologías para este proyecto se seleccionó por medio de la siguiente pregunta:

¿Cómo impacta la implementación de mejoras en la infraestructura de una red tipo GRID en términos de eficiencia, capacidad de procesamiento y rendimiento socioeconómico?

- **Análisis y Diagnóstico:** Esta etapa implica una evaluación minuciosa de la red GRID existente, llevada a cabo mediante la recopilación de datos y el análisis detallado de la infraestructura, configuraciones, protocolos de seguridad y métricas de rendimiento. Se utilizarán herramientas de monitoreo y diagnóstico para identificar posibles deficiencias, cuellos de botella y áreas de mejora. El objetivo es obtener una comprensión integral del estado actual de la red y de las limitaciones que afectan su eficiencia y capacidad de procesamiento.
- **Diseño y Desarrollo:** En esta fase, se elaborará un plan de actualización detallado basado en los hallazgos del análisis previo. Se diseñarán soluciones específicas para abordar las deficiencias identificadas, centrándose en aspectos clave como la arquitectura de red, la seguridad de los datos, la escalabilidad y la disponibilidad. Se utilizarán mejores prácticas y estándares de la industria para garantizar la eficacia y la viabilidad de las soluciones propuestas.
- **Implementación:** Una vez completado el diseño, se procederá a la implementación de las mejoras planificadas en la red GRID. Esto puede implicar la actualización de hardware y software, la configuración de

nuevos componentes de red, la aplicación de parches de seguridad y la integración de nuevas tecnologías. Se seguirán procedimientos rigurosos de implementación para minimizar el impacto en la operatividad del sistema y garantizar una transición fluida hacia la nueva infraestructura.

- Pruebas y Evaluación: Se llevarán a cabo pruebas exhaustivas de rendimiento y eficiencia para evaluar el impacto de las mejoras implementadas en la red GRID. Se utilizarán casos de prueba realistas y escenarios de carga para medir parámetros como el tiempo de respuesta, la utilización de recursos y la capacidad de escalabilidad. Los resultados obtenidos se compararán con las métricas de rendimiento de la red original para determinar la efectividad de las mejoras realizadas.
- Análisis de Impacto Socioeconómico: En esta etapa, se evaluará el impacto socioeconómico de la red optimizada, analizando cómo las mejoras contribuyen a la reducción de costos operativos, tiempos de procesamiento y recursos requeridos. Se llevará a cabo un análisis detallado de los beneficios tangibles e intangibles derivados de la actualización de la red, con el objetivo de cuantificar el retorno de la inversión y demostrar el valor agregado de las mejoras realizadas.
- Facetas de la metodología :
 - Fase 1: Preparación y Análisis
 - Definición detallada del proyecto.
 - Revisión de la literatura sobre redes tipo GRID y tecnologías relacionadas.
 - Identificación y selección de la red GRID a ser optimizada.
 - Análisis de la infraestructura existente y diagnóstico de limitaciones.
 - Fase 2: Diseño y Planificación

- Diseño de las mejoras en la arquitectura de la red.
- Planificación de las etapas de actualización.
- Diseño de medidas de seguridad mejoradas.
- Investigación y selección de soluciones de escalabilidad.
- Creación del plan de implementación detallado.
- Obtención de los recursos necesarios para la actualización.
- Fase 3: Implementación
 - Implementación de mejoras en la arquitectura de la red.
 - Integración de medidas de seguridad y escalabilidad.
 - Configuración de los sistemas y componentes actualizados.
 - Realización de pruebas preliminares en entorno controlado.
- Fase 4: Pruebas y Evaluación
 - Ejecución de pruebas de rendimiento y eficiencia.
 - Recopilación de datos y métricas para análisis.
 - Comparación de los resultados con la red original.
 - Ajustes y optimizaciones adicionales según los hallazgos.
- Fase 5: Análisis de Impacto y Conclusiones
 - Evaluación del impacto socioeconómico de la red optimizada.
 - Comparativa de costos y beneficios con la situación anterior.
 - Redacción del informe final de tesis.
 - Conclusiones y recomendaciones basadas en los resultados obtenidos.
- Fase 6: Presentación y Defensa
 - Preparación de la presentación de la tesis.
 - Preparación para la defensa oral ante el comité evaluador.
 - Presentación y defensa de la tesis.
 - Respuestas a preguntas y discusión con el comité evaluador.

- Fase 7: Cierre y Entrega
 - Revisión final del informe de tesis.
 - Realización de las correcciones finales sugeridas por el comité evaluador.
 - Entrega de la versión final del informe de tesis.
 - Cierre administrativo del proyecto.

1.6 Infraestructura.

La supercomputadora Cuetlaxcoapan del LNS está compuesta por un clúster estándar de cálculo con procesadores Intel Xeon y un clúster con procesadores Intel Xeon Phi Knights Landing. Las características del clúster Intel Xeon son:

- 228 nodos de cálculo Thin (5472 núcleos totales)
Cada nodo contiene:
 - 2 procesadores Intel Xeon E5-2680 v3 (Haswell) a 2.5 GHz
 - 12 núcleos por procesador / 24 núcleos totales
 - 128 GB de memoria RAM DDR4 a 2133 MHz
 - 2 interfaces de red Ethernet Gigabit
 - interface de red InfiniBand FDR a 56 Gbps
- 2 nodos de cálculo con GPU (48 núcleos CPU y 11520 núcleos CUDA totales)
Cada nodo contiene:
 - 2 procesadores Intel Xeon E5-2680 v3 (Haswell) a 2.5 GHz por nodo
 - 12 núcleos por procesador / 24 núcleos totales
 - 128 GB de memoria RAM DDR4 a 2133 MHz
 - 2 interfaces de red Ethernet Gigabit
 - 1 interface de red InfiniBand FDR a 56 Gbps
 - 2 tarjetas GPU NVIDIA Tesla K40
 - 2880 núcleos CUDA por GPU / 5769 núcleos CUDA totales
 - 12 GB de memoria RAM DDR5 por GPU
 - Almacenamiento:

Se dispone de 1.2 PB de espacio total para almacenamiento en disco

- Clúster:

Compuesto por:

- 6 servidores OSS
- 2 servidores MDS
- 1 servidor de almacenamiento MDT de 40 TB
- 3 servidores de almacenamiento OST de 320 TB cada uno

Redes de Comunicaciones. Se dispone de:

- una red Ethernet Gigabit para gestión del hardware de la supercomputadora y el aprovisionamiento de software a los nodos. Una red Infiniband FDR (56 Gpbs) para comunicación rápida entre nodos de cálculo

La infraestructura de la supercomputadora Cuetlaxcoapan del LNS es impresionante y está diseñada para proporcionar un rendimiento de vanguardia en el ámbito del cálculo intensivo. Consta de un clúster estándar de cálculo con procesadores Intel Xeon y un clúster con procesadores Intel Xeon Phi Knights Landing.

El clúster Intel Xeon presenta 228 nodos de cálculo Thin, lo que suma un total de 5472 núcleos. Cada nodo contiene dos procesadores Intel Xeon E5-2680 v3 (Haswell) con 12 núcleos cada uno, alcanzando un total de 24 núcleos. Además, cuentan con 128 GB de memoria RAM DDR4 a 2133 MHz y dos interfaces de red Ethernet Gigabit, así como una interfaz de red InfiniBand FDR a 56 Gbps.

Además, hay dos nodos de cálculo con GPU, cada uno con 48 núcleos CPU y 11520 núcleos CUDA totales. Estos nodos también tienen dos procesadores Intel Xeon E5-2680 v3, 128 GB de RAM DDR4, y la misma configuración de red que los nodos de cálculo estándar. Sin embargo, lo más destacado son las dos tarjetas GPU NVIDIA Tesla K40 en cada nodo, cada una con 2880 núcleos CUDA y 12 GB de memoria RAM DDR5.

El almacenamiento es igualmente impresionante, con 1.2 PB de espacio total en disco, distribuido en servidores OSS, servidores MDS, y servidores de almacenamiento MDT y OST.

En cuanto a las redes de comunicaciones, se dispone de una red Ethernet Gigabit para la gestión del hardware y el aprovisionamiento de software, así como una red Infiniband FDR de 56 Gpbs para una comunicación rápida entre los nodos de cálculo.

En resumen, la infraestructura de la supercomputadora Cuetlaxcoapan del LNS es robusta y completa, ofreciendo un entorno ideal para el procesamiento de grandes cantidades de datos y cálculos intensivos en un tiempo récord.

Con las prácticas modernas de las tecnologías de la información lo que se obtuvo promover el ahorro de los costos en desarrollar proceso y estándares para el desarrollo de un proyecto de investigación en los centros de investigación, instituciones educativas o empresas de manera abierta, como una alternativa el cual eleva los costos considerablemente y a su vez se necesita un personal especializado que busquen constantemente nueva información de los tópicos contemporáneos, diseñando e implementando medidas y políticas que cierren dichas brechas, de manera constante y costosa, además de poder brindarles a los alumnos de posgrado o usuarios de instituciones de investigación científica, un ejemplo real de cómo se maneja la investigación de proyectos importantes en ciencias y tecnologías de la información y comunicación, haciendo más fácil la transición de estudiante a colaborador en un proyecto individual o grupal de investigación de un centro de investigación o universidad.

Con esta investigación y creación de la documentación, el proceso para el desarrollo se utilizará en los centros de investigación, instituciones educativas o empresas para no tener pérdidas de información. Siguiendo estándares que garanticen de manera cualitativa el mejor manejo de la información, así como presentar esta investigación en un congreso de tecnologías de la información en la sección de calidad y por último plasmar toda la investigación en un documento de tesis que será accesible a toda la comunidad científica y tecnológica.

2 Estado del arte.

En el campo de las redes de cómputo distribuido [4], particularmente las redes tipo GRID, se han realizado diversos avances y desarrollos con el propósito de optimizar su rendimiento y eficiencia. A continuación, se presentan algunos de los trabajos más relevantes que abordan aspectos similares al proyecto de tesis "Infraestructura y Actualización de una Red tipo GRID".

Como se menciona en [5], la configuración y preparación adecuadas son cruciales para implementar con éxito Oracle Grid Infrastructure y Oracle RAC. Estos componentes permiten el procesamiento distribuido y de alta disponibilidad en el ámbito de las bases de datos. El proceso de instalación implica una serie de etapas, cada una con requisitos específicos que garantizan la funcionalidad óptima del sistema. A continuación, se presenta un análisis de los elementos clave involucrados en la configuración y preparación de Oracle Grid Infrastructure y Oracle RAC.

El primer paso en la implementación de Oracle Grid Infrastructure implica verificar el hardware del servidor de instalación. Esta verificación asegura que el hardware cumpla con los requisitos mínimos para soportar el sistema [6, 7]. Además, se aborda la configuración del entorno, incluyendo aspectos de red, como la configuración de la red pública utilizada por Oracle Grid Infrastructure y Oracle RAC [8].

La preparación de servidores involucra la evaluación y cumplimiento de requisitos de hardware y almacenamiento [6]. Esto incluye verificar la memoria y el almacenamiento mínimo necesario para el servidor. Los requisitos

específicos para sistemas de 64 bits y las versiones compatibles de sistemas operativos también deben ser tenidos en cuenta para asegurar un funcionamiento fluido [9].

Para garantizar la coherencia y funcionalidad del clúster, es esencial preparar adecuadamente los nodos individuales [10]. Esto abarca aspectos como la creación de directorios de inventario y de inicio, así como la sincronización de tiempo en todos los nodos. La correcta configuración de cuentas de usuario y la preparación del software de Oracle también son etapas fundamentales en este proceso.

La instalación de Oracle Grid Infrastructure y Oracle RAC implica varios pasos cruciales. Desde la instalación de la infraestructura de Oracle Grid [11] hasta la creación de grupos de discos a través de ASMCA [12], cada fase contribuye a la creación de un entorno robusto y altamente disponible.

Finalmente, la instalación del software RDBMS completa el proceso de configuración [13]. Esta etapa implica la instalación del motor de base de datos Oracle en los nodos preparados previamente. Asegurarse de que los requisitos del software del sistema operativo sean satisfechos es esencial para lograr una instalación exitosa [14].

Estos estudios abordan diferentes aspectos relacionados con la optimización de redes Grid, incluyendo la asignación dinámica de recursos, mejoras de seguridad, desafíos de escalabilidad y análisis económicos de actualizaciones [14-16]. La revisión de estos trabajos permitirá situar el proyecto actual en el contexto más amplio de la investigación en redes distribuidas y ayudará a identificar áreas específicas en las que se pueden aportar nuevas contribuciones.

3 Configuración de una red GRID de Oracle rac 2cr1 (12.1.0.2) en solaris sparc-64

La Tabla 1 proporciona una lista de verificación detallada para el hardware del servidor necesario para la instalación de Oracle Grid Infrastructure. Esta lista incluye varios elementos clave. Primero, se debe confirmar que las marcas, modelos y arquitectura central del servidor, así como los adaptadores de bus de host (HBA), sean compatibles con Oracle Grid Infrastructure y Oracle RAC, garantizando así que el hardware pueda soportar adecuadamente el software de Oracle (Oracle, 2021). El nivel de ejecución recomendado es 3, asegurando que el sistema esté en un estado adecuado para la instalación. Además, es esencial que las tarjetas de visualización del servidor tengan una resolución mínima de 1024 x 768 para el Oracle Universal Installer, lo que asegura que el monitor de visualización sea adecuado para la instalación. En cuanto a la memoria, se requiere al menos 8 GB de RAM para las instalaciones de Oracle Grid Infrastructure, proporcionando la capacidad necesaria para ejecutar los procesos del sistema. Finalmente, las interfaces de gestión de plataforma inteligente (IPMI) deben estar instaladas y configuradas correctamente, y se debe disponer de la información de la cuenta de administrador de IPMI para la persona que realiza la instalación. Asegurarse de que las interfaces del controlador de administración de la placa base (BMC) estén configuradas y tengan las credenciales necesarias disponibles es crucial para el proceso de instalación (Oracle, 2021).

Tabla 1. Lista de verificación de hardware de servidor para Oracle Grid Infrastructure

VERIFICAR	TAREA
Marca y arquitectura del servidor	Confirme que las marcas, los modelos, la arquitectura central y los adaptadores de bus de host (HBA) del servidor sean compatibles para ejecutarse con Oracle Grid Infrastructure y Oracle RAC.
Nivel de ejecución	3
Tarjetas de visualización del servidor	Resolución de pantalla de al menos 1024 x 768 para Oracle Universal Installer. Confirme el monitor de visualización.
Memoria mínima de acceso aleatorio (RAM)	Al menos 8 GB de RAM para instalaciones de Oracle Grid Infrastructure
Interfaz de gestión de plataforma inteligente (IPMI)	Tarjetas IPMI instaladas y configuradas, con información de la cuenta de administrador de IPMI disponible para la persona que ejecuta la instalación. Asegúrese de que las interfaces del controlador de administración de la placa base (BMC) estén configuradas y tengan un nombre de usuario y contraseña de cuenta de administración para proporcionar cuando se le solicite durante la instalación.

La Tabla 2 detalla los requisitos del sistema operativo necesarios para la instalación de Oracle Grid Infrastructure en Oracle Solaris. En primer lugar, es necesario que Open SSH esté instalado manualmente si no se incluyó en la

instalación predeterminada de Oracle Solaris, asegurando así una conexión segura y cifrada para la administración del sistema (Oracle, 2021). En cuanto a la compatibilidad de los núcleos de Oracle Solaris en SPARC (64 bits), se soportan versiones como Oracle Solaris 11.4 (Oracle Solaris 11.4.0.0.1.15.0) o posteriores SRU, Oracle Solaris 11.3 SRU 7.6 (Oracle Solaris 11.3.7.6.0) o SRU posteriores, y Oracle Solaris 11.2 SRU 5.5 (Oracle Solaris 11.2.5.5.0) o SRU posteriores, además de Oracle Solaris 10 Actualización 11 (Oracle Solaris 10 1/13s10s_u11wos_24a) o actualizaciones posteriores. De manera similar, los núcleos de Oracle Solaris en x86-64 (64 bits) que son compatibles incluyen Oracle Solaris 11.4 (Oracle Solaris 11.4.0.0.1.15.0) o posteriores SRU, Oracle Solaris 11.3 SRU 7.6 (Oracle Solaris 11.3.7.6.0) o SRU posteriores, y Oracle Solaris 11.2 SRU 5.5 (Oracle Solaris 11.2.5.5.0) o SRU posteriores, junto con Oracle Solaris 10 Actualización 11 (Oracle Solaris 10 1/13s10x_u11wos_24a) o actualizaciones posteriores. Estos requisitos aseguran que el sistema operativo esté optimizado y sea compatible con Oracle Grid Infrastructure, proporcionando un entorno estable y eficiente para su funcionamiento (Oracle, 2021).

Tabla 2. Lista de verificación del sistema operativo para oracle grid infrastructure en oracle solaris

REQUERIMIENTO	TAREA
Sistema Operativo requerimientos generales	Open SSH instalado manualmente, si aún no lo tienes instalado como parte de una instalación predeterminada de Oracle Solaris
	Los siguientes núcleos de Oracle Solaris en SPARC (64 bits) soportan: Oracle Solaris 11.4 (Oracle Solaris 11.4.0.0.1.15.0) o

	<p>posterior SRU Oracle Solaris 11.3 SRU 7.6 (Oracle Solaris 11.3.7.6.0) o SRU posteriores a Oracle Solaris 11.2 SRU 5.5 (Oracle Solaris 11.2.5.5.0) o SRU posteriores Oracle Solaris 10 Actualización 11 (Oracle Solaris 10 1/13s10s_u11wos_24a) o actualizaciones posteriores.</p>
	<p>Los siguientes kernels de Oracle Solaris en x86-64 (64 bits) soportan: Oracle Solaris 11.4 (Oracle Solaris 11.4.0.0.1.15.0) o posterior SRU Oracle Solaris 11.3 SRU 7.6 (Oracle Solaris 11.3.7.6.0) o SRU posteriores Oracle Solaris 11.2 SRU 5.5 (Oracle Solaris 11.2.5.5.0) o SRU posteriores Oracle Solaris 10 Actualización 11 (Oracle Solaris 10 1/13s10x_u11wos_24a) o actualizaciones posteriores.</p>

La Tabla 3 presenta una lista de verificación para la configuración del servidor necesaria para implementar Oracle Grid Infrastructure. Primero, se debe asegurar que el sistema de archivos temporal tenga al menos 1 GB de espacio, aunque Oracle recomienda 2 GB o más para un funcionamiento óptimo. En cuanto a la asignación de espacio de intercambio (swap) en relación con la RAM, si la RAM está entre 4 GB y 16 GB, el espacio de intercambio debe ser igual al tamaño de la RAM; para más de 16 GB de RAM, se requiere 16 GB de espacio de intercambio. Además, se recomienda crear una configuración óptima y flexible para las rutas de los puntos de montaje de los binarios del software,

asegurándose de que la ruta de inicio de Oracle utilice solo caracteres ASCII, incluyendo nombres de usuarios y directorios, para evitar problemas durante la instalación y operación.

También es crucial establecer un protocolo de hora de red para la sincronización del tiempo del grupo. Oracle Clusterware requiere que todos los nodos del clúster tengan la misma configuración de zona horaria y variables de entorno, lo cual puede lograrse mediante la configuración de un protocolo de tiempo de red (NTP) o el servicio Oracle Cluster TimeService. Esta sincronización es fundamental para garantizar la coherencia y el correcto funcionamiento de los componentes del clúster.

Tabla 3. Lista de verificación de configuración del servidor para Oracle Grid Infrastructure

VERIFICAR	TAREA
Espacio en disco asignado al sistema de archivos temporal	Al menos 1 GB de espacio en el directorio temporal, Oracle recomienda 2 GB o más
Intercambiar asignación de espacio en relación con la RAM	Entre 4 GB y 16 GB: Igual al tamaño de la RAM, Más de 16 GB: 16 GB
Rutas de los puntos de montaje para los binarios del software	Oracle recomienda crear un Optimal Flexible. Configuración de arquitectura
Asegúrese de que el inicio de Oracle (la ruta de inicio de Oracle que seleccione para la base de datos Oracle) utilice solo caracteres ASCII.	La restricción de caracteres ASCII incluye al usuario propietario de la instalación, nombres, que se utilizan de forma predeterminada para algunas rutas de inicio, como, así como otros nombres de directorio que puede seleccionar para las rutas.
Establecer protocolo de hora de red para la hora del grupo. Sincronización	Oracle Clusterware requiere el mismo entorno de zona horaria, configuración de

	<p>variables en todos los nodos del clúster. Asegúrese de configurar la sincronización de zona horaria en todos los nodos del clúster que utilizan un sistema operativo configurado y un protocolo de tiempo de red (NTP) u Oracle Cluster TimeService de sincronización.</p>
--	---

La Tabla 4 detalla una lista de verificación para la configuración de red necesaria al implementar Oracle Grid Infrastructure. En cuanto al hardware de red pública, se recomienda utilizar conmutadores de red públicos redundantes conectados a una puerta de enlace pública y a los puertos de interfaz pública de cada nodo del clúster. Además, se deben emplear tarjetas de interfaz Ethernet redundantes y troncalizadas para asegurar una conectividad estable y confiable. Los conmutadores y las interfaces de red deben soportar al menos 1 Gb de ancho de banda, y el protocolo de red utilizado debe ser TCP/IP.

Para la red privada de interconexión entre los nodos del clúster, se aconseja utilizar conmutadores de red privados dedicados, también redundantes, conectados a los puertos de interfaz privados de cada nodo. Estos conmutadores y adaptadores de interfaz de red deben tener capacidad mínima de 1 GbE, y la interconexión debe ser compatible con el protocolo UDP (User Datagram Protocol). Oracle recomienda la configuración de Jumbo Frames (tramas gigantes Ethernet de más de 1500 bits) para mejorar la eficiencia de la transmisión UDP, aunque es esencial realizar pruebas de carga del sistema para garantizar su correcto funcionamiento y habilitación en toda la pila de red.

Esta configuración de red asegura una infraestructura robusta y optimizada para soportar las demandas de Oracle Grid Infrastructure, garantizando tanto la disponibilidad como el rendimiento necesario para entornos de procesamiento distribuido y alta disponibilidad.

Tabla 4. Lista de verificación de red para Oracle Grid Infrastructure

VERIFICAR	TAREA
Hardware de red pública	<ul style="list-style-type: none"> • Conmutador de red pública (se recomiendan conmutadores redundantes) conectado a una puerta de enlace pública y a los puertos de interfaz pública para cada nodo miembro del clúster. • Tarjeta de interfaz Ethernet (tarjetas de red redundantes recomendado, troncalizado como un nombre de puerto Ethernet). • Los conmutadores y las interfaces de red deben ser de al menos 1 Gb. • El protocolo de red es el Protocolo de control de transmisión (TCP) y Protocolo de Internet (IP).
Hardware de red privada para la interconexión.	<ul style="list-style-type: none"> • Conmutadores de red dedicados privados (conmutadores redundantes recomendados), conectado a los puertos de interfaz privados para cada nodo miembro del clúster. • Los conmutadores y adaptadores de interfaz de red deben tener al menos 1GbE. • La interconexión debe admitir el

	<p>protocolo de datagramas de usuario (UDP).</p> <ul style="list-style-type: none"> • Las tramas gigantes (tramas Ethernet de más de 1500 bits) como un estándar IEEE, pero puede reducir la sobrecarga de UDP si se configura, Oracle recomienda el uso de Jumbo Frames para interconexión. Sin embargo, tenga en cuenta que debe realizar una prueba de carga del sistema y asegúrese de que estén habilitados en toda la pila
--	---

La Tabla 5 presenta una lista detallada de verificación para la configuración del entorno de usuario necesario al implementar la infraestructura de red de Oracle. A continuación, se explica cada tarea:

Revisar Oracle Inventory y OINSTALL: El directorio Oracle Inventory es crucial como el inventario central del software Oracle instalado en el sistema. Debe ser el grupo principal para todos los propietarios de instalaciones de software de Oracle. Los usuarios que tienen este grupo como principal obtienen el privilegio OINSTALL, que les permite leer y escribir en el inventario central.

Crear Grupos de Sistemas Operativos y Usuarios para Privilegios de Sistema: Se deben establecer configuraciones de límites de recursos y otros requisitos para los propietarios de instalaciones de software de Oracle. Esto garantiza que los usuarios tengan los privilegios adecuados asignados por roles para gestionar eficazmente las instalaciones de Oracle.

Desconfigurar Software de Oracle: Si existe una instalación previa de software de Oracle y se utiliza el mismo usuario para la nueva instalación, es necesario desactivar variables de entorno específicas como \$ORACLE_HOME, \$ORA_NLS10 y \$TNS_ADMIN. Esto evita conflictos y asegura que la nueva instalación se realice de manera limpia y adecuada.

VARIABLES DE ENTORNO: Configurar correctamente las variables de entorno es fundamental. Específicamente, se debe establecer la variable de entorno DISPLAY adecuadamente para asegurar la correcta visualización de interfaces gráficas durante la configuración y gestión del software Oracle.

Configurar el Entorno del Propietario: Para Oracle o Grid, es esencial configurar el entorno del usuario adecuadamente. Esto incluye establecer la máscara de creación del modo de archivo predeterminado en 022 en el archivo de inicio del shell para asegurar permisos correctos de archivo. Además, configurar la variable de entorno DISPLAY es crucial para la interfaz gráfica de usuario.

Determinar la Opción de Delegación de Privilegios de Root: Durante la instalación, se requerirá ejecutar scripts de configuración como usuario root. Es vital determinar la opción de delegación de privilegios de root de manera adecuada, ya sea utilizando las credenciales de usuario root directamente o mediante el uso de sudo para ejecutar comandos con privilegios elevados de manera segura.

Esta tabla proporciona una guía detallada para configurar correctamente el entorno de usuario necesario para la infraestructura de red de Oracle, asegurando que todas las configuraciones y privilegios estén establecidos de manera adecuada para una operación óptima y segura del software Oracle.

Tabla 5. Lista de verificación de la configuración del entorno de usuario para la infraestructura de red de Oracle

VERIFICAR	TAREA
<p>Revisar Oracle Inventory y OINSTALL Requerimientos de Grupo</p>	<p>El directorio de Oracle Inventory es el inventario central del software Oracle instalado en su sistema. Debería ser el grupo principal para todos los propietarios de instalaciones de software de Oracle. A los usuarios que tienen el grupo Oracle Inventory como grupo principal se les concede el privilegio OINSTALL para leer y escribir en el inventario central.</p>
<p>Cree grupos de sistemas operativos y usuarios para privilegios de sistema estándar o asignados por roles.</p>	<p>Establezca configuraciones de límites de recursos y otros requisitos para los propietarios de instalaciones de software de Oracle.</p>
<p>Desconfigurar software de Oracle Variables de entorno</p>	<p>Si tiene una instalación de software de Oracle existente y está utilizando el mismo usuario para instalar esta instalación, desactive las siguientes variables de entorno: \$ORACLE_HOME; \$ORA_NLS10; \$TNS_ADMIN</p>
<p>Configurar el software de Oracle Entorno del propietario</p>	<p>Configure el entorno del usuario de Oracle o Grid realizando las siguientes tareas:</p> <ul style="list-style-type: none"> • Establezca la máscara de creación del modo de archivo predeterminado en 022 en el archivo de inicio del shell. • Establecer la variable de entorno DISPLAY

<p>Determinar la opción de delegación de privilegios de root para la instalación.</p>	<p>Durante la instalación, se le solicita que ejecute scripts de configuración como usuario root</p> <ul style="list-style-type: none"> • Utilice credenciales de usuario raíz • Utilice sudo
---	---

La Tabla 6 presenta una lista de verificación detallada para la configuración de almacenamiento necesaria al implementar la infraestructura de red de Oracle. A continuación, se explica cada tarea:

1. Espacio Mínimo en Disco para la Infraestructura de Red de Oracle:

- Se requieren al menos 12 GB de espacio para Oracle Grid Infrastructure (Grid home) para un clúster.
- Además, se debe reservar al menos 9 GB para la Oracle Database Enterprise Edition, asegurando que haya suficiente espacio disponible para la instalación y operación del software.

2. Seleccione Oracle ASM:

- Durante la instalación, se le solicitará seleccionar Oracle Automatic Storage Management (ASM), que es una solución integrada y altamente disponible para la gestión de almacenamiento.
- Deberá proporcionar rutas de almacenamiento específicas para los archivos de Oracle Clusterware según la configuración del clúster.

3. Opciones de Almacenamiento:

- Antes de instalar Oracle Grid Infrastructure, es necesario crear un archivo de manifiesto que especifique los detalles de almacenamiento, incluidas

las configuraciones de Oracle ASM para asegurar la correcta configuración y operación del clúster.

4. Seleccionar el Repositorio de Gestión de la Infraestructura de Red (GIMR):

- Dependiendo del tipo de clúster que se esté instalando, se puede elegir alojar el Repositorio de Gestión de Infraestructura de Red (GIMR) dentro del mismo clúster o en un clúster remoto.
- Los detalles de GIMR deben especificarse en el archivo de manifiesto del clúster miembro antes de la instalación para garantizar una gestión eficiente y centralizada de la infraestructura de red.

5. Eliminar el Cifrado del Archivo ACFS de Oracle:

- Antes de la instalación, es crucial eliminar cualquier cifrado aplicado a los sistemas de archivos Oracle ACFS (Automatic Storage Management Cluster File System) para evitar posibles problemas de corrupción de datos durante la operación.

Esta tabla proporciona una guía exhaustiva para configurar correctamente el almacenamiento necesario para la infraestructura de red de Oracle, asegurando que se cumplan todos los requisitos y configuraciones recomendadas para una implementación exitosa y eficiente del software Oracle.

Tabla 6. Lista de verificación de almacenamiento para la infraestructura de red de Oracle

VERIFICAR	TAREA
Espacio mínimo en disco para la infraestructura de red de Oracle	<ul style="list-style-type: none">• 12 GB de espacio para Oracle Grid Infrastructure para un hogar de

Software	<p>clúster (Grid home).</p> <ul style="list-style-type: none"> Al menos 9 GB para Oracle Database Enterprise Edition
<p>Seleccione Oracle ASM</p> <p>Opciones de Almacenamiento</p>	<p>Durante la instalación, según la configuración del clúster, se le solicita que proporcione rutas de almacenamiento de Oracle ASM para los archivos de Oracle Clusterware.</p> <p>Antes de instalar Oracle Member Cluster, cree un archivo de manifiesto que especifique los detalles de almacenamiento.</p>
<p>Seleccionar el repositorio de gestión de la infraestructura de red (GIMR)</p> <p>Opción de almacenamiento</p>	<p>Dependiendo del tipo de clúster que esté instalando, puede elegir alojar el Repositorio de administración de infraestructura de red (GIMR) para un clúster en el mismo clúster o en un clúster remoto</p> <p>Debe especificar los detalles de GIMR cuando crea el archivo de manifiesto del clúster miembro antes de la instalación.</p>
<p>Eliminar el cifrado del archivo ACFS de Oracle</p> <p>Sistemas antes de la instalación</p>	<p>Para evitar la corrupción de datos, asegúrese de eliminar el cifrado de los sistemas de archivos Oracle ACFS antes de la instalación.</p>

La Tabla 7 presenta una lista detallada de verificación para la configuración de almacenamiento necesaria al implementar la Infraestructura de Red de Oracle.

A continuación se explica cada tarea:

Espacio Mínimo en Disco para la Infraestructura de Red de Oracle:

Se requieren al menos 12 GB de espacio para Oracle Grid Infrastructure (Grid home) por cada instalación de clúster. Además, se recomienda reservar al menos 9 GB para la Oracle Database Enterprise Edition, asegurando así suficiente espacio para la instalación y operación del software.

Seleccione Oracle ASM:

Durante el proceso de instalación, se le pedirá seleccionar Oracle Automatic Storage Management (ASM), una solución integrada y altamente disponible para la gestión de almacenamiento. Deberá especificar las rutas de almacenamiento para los archivos de Oracle Clusterware según la configuración del clúster.

Opciones de Almacenamiento:

Antes de instalar Oracle Grid Infrastructure, es fundamental crear un archivo de manifiesto que detalle las configuraciones de almacenamiento, incluidas las ajustes de Oracle ASM, para garantizar una configuración adecuada y un funcionamiento correcto del clúster.

Seleccione el Repositorio de Gestión de la Infraestructura de Red (GIMR):

Según el tipo de clúster que se esté instalando, se puede optar por alojar el Repositorio de Gestión de Infraestructura de Red (GIMR) dentro del mismo

clúster o en uno remoto. Los detalles del GIMR deben especificarse en el archivo de manifiesto del miembro del clúster antes de la instalación para asegurar una gestión centralizada y eficiente de la infraestructura.

Eliminar el Cifrado del Archivo ACFS de Oracle:

Antes de la instalación, es crucial eliminar cualquier cifrado aplicado a los sistemas de archivos Oracle ACFS (Automatic Storage Management Cluster File System) para evitar posibles problemas de corrupción de datos durante la operación.

Esta tabla proporciona una guía exhaustiva para configurar correctamente la infraestructura de almacenamiento necesaria para Oracle Grid Infrastructure, asegurando el cumplimiento de todos los requisitos y configuraciones recomendadas para una implementación exitosa y eficiente del software de Oracle.

Tabla 7. Lista de verificación de implementación de clústeres para la infraestructura de red de Oracle

VERIFICAR	TAREA
Configuración del clúster de Oracle que alberga todos los servicios localmente y accede al almacenamiento directamente de la infraestructura de red de Oracle y Oracle ASM	Implementar un clúster independiente de Oracle. Utilice la opción Oracle Extended Clúster para ampliar un clúster de Oracle RAC en dos o más sitios separados, cada uno equipado con su propio almacenamiento.
Configuración de un clúster de dominio de Oracle para estandarizar, centralizar y optimizar su clúster de aplicación real de	Implementar un clúster de servicios de dominio de Oracle. Para ejecutar Oracle Real Application

Oracle (Oracle RAC).	Clusters (Oracle RAC) u instancias de base de datos Oracle RAC One Node, implemente Oracle Member Cluster para bases de datos Oracle. Para ejecutar aplicaciones de software de alta disponibilidad, implemente Oracle Member Cluster for Applications
----------------------	---

3.1 Configuración automática de Oracle Solaris con Oracle Database.

Se debe utilizar el paquete de grupo de requisitos previos de Oracle Database para simplificar la configuración del sistema operativo Oracle Solaris como preparación para la instalación del software de Oracle.

Oracle recomienda instalar el paquete de grupo de requisitos previos de Oracle Database `oracle-rdbms-server-12-1-preinstall` para preparar las instalaciones de Oracle Database y Oracle Grid Infrastructure.

Pasos:

- Acerca de los paquetes de requisitos previos de Oracle Database para Oracle Solaris: Se utiliza el grupo de paquetes de requisitos previos de Oracle Database para simplificar la configuración del sistema operativo y asegurarse de que dispone de los paquetes necesarios.
- Comprobación de la configuración de los paquetes de requisitos previos de Oracle Database: Se utiliza este procedimiento para recopilar información sobre la configuración del paquete de grupo de requisitos previos de Oracle Database.

- Instalación de los paquetes de requisitos previos de Oracle Database para Oracle Solaris. Se utiliza este procedimiento para instalar el paquete de grupo de requisitos previos de Oracle Database para su software Oracle.

Acerca de los paquetes de requisitos previos de Oracle Database para Oracle Solaris, se puede instalar `oracle-rdbms-server-12-1-preinstall` aunque haya instalado Oracle Solaris utilizando cualquiera de los grupos de paquetes de servidor, como `solaris-minimal-server` `solaris-small-server`, `solaris-large-server` o `solaris-desktop`.

La configuración de un servidor mediante Oracle Solaris y los requisitos previos de Oracle Database consiste en los siguientes pasos:

- Instale la versión de Oracle Solaris recomendada para Oracle Database.
- Instale el paquete del grupo de requisitos previos de Oracle Database `oracle-rdbmsserver-12-1-preinstall`.
- Cree grupos y usuarios con asignación de funciones.
- Complete la configuración de la interfaz de red para cada nodo candidato a cluster.
- Complete la configuración del sistema para el acceso al almacenamiento compartido según sea necesario para cada candidato a clúster de nodo estándar o núcleo.

Una vez completados estos pasos, se puede instalar Oracle Database, Oracle Grid Infrastructure u Oracle RAC.

3.2 Comprobación de la configuración de los paquetes de requisitos previos de Oracle Database

Se utiliza este procedimiento para recopilar información sobre la configuración de los paquetes del grupo de requisitos previos de Oracle Database de Oracle Database.

- Para comprobar si oracle-rdbms-server-12-1-preinstall ya está instalado:

```
$ pkg list oracle-rdbms-server-12-1-preinstall
```

- Para comprobar si existe la última versión de oracle-rdbms-server-12-1-preinstall:

```
$ pkg list -n oracle-rdbms-server-12-1-preinstall
```

- Antes de instalar oracle-rdbms-server-12-1-preinstall:

1. Utilice la opción -n para comprobar si hay errores:\$

```
pkg install -nv oracle-rdbms-server-12-1-preinstall
```

Nota: No tiene que especificar el nombre completo del paquete, sólo la parte final del nombre que sea única.

2. Si no hay errores, inicia sesión como root e instala el paquete de grupo:

```
# pkg install oracle-rdbms-server-12-1-preinstall
```

- Para ver los paquetes instalados por oracle-rdbms-server-12-1-preinstall:

```
$ pkg contents -ro type,fmri -t depend oracle-rdbms-server-12-1-preinstall
```

La instalación de los paquetes de requisitos previos de Oracle Database para Oracle Solaris. Se utiliza este procedimiento para instalar el paquete de grupo de requisitos previos de Oracle Database para su software Oracle.

El paquete de group/prerequisite/oracle/oracle-rdbms-server-12-1-preinstall instala todos los paquetes necesarios para la instalación de Oracle Database y Oracle Grid y Oracle Grid Infrastructure.

Para instalar los paquetes de grupo oracle-rdbms-server-12-1-preinstall, inicie sesión como root y ejecute el siguiente comando en sistemas Oracle Solaris 11.2.5.5.0 y posteriores:

```
# pkg install oracle-rdbms-server-12-1-preinstall
```

Comprobación de requisitos de hardware para Oracle Grid

Comprobación de requisitos:

- Verificar que las especificaciones de los servidores cumplen los requisitos mínimos recomendados por Oracle para la instalación.
- Confirmar que hay suficiente RAM instalada, espacio de swap configurado, y espacio en disco en /tmp.
- Esta sección detalla los requerimientos específicos de CPU, RAM, disco duro y particionamiento.
- No cubre recomendaciones de optimización de recursos o configuraciones para cargas de trabajo específicas.

Inicio de sesión remoto con X Window System:

- Permite ejecutar la interfaz gráfica de Oracle Universal Installer (OUI) en un sistema remoto Linux.
- Útil cuando los ajustes de ese sistema remoto prohíben el acceso directo a interfaz gráfica.
- Redirige y despliega la pantalla GUI en un sistema cliente Windows o Linux conectado.
- Comandos: xhost + para agregar y habilitar acceso desde el sistema remoto, ssh -Y para conectar con X11 habilitado

Comandos para comprobación de hardware y memoria:

- `sar -r`: muestra páginas de memoria física y bloques de swap no utilizados actualmente.
- `swap -s`: muestra uso actual y tamaño total configurado para espacio de swap.
- `df -kh /tmp`: muestra espacio libre disponible en directorio `/tmp`
- `df -kh`: muestra todo el espacio libre en disco en formato legible.
- `prtconf`: muestra cantidad de memoria RAM instalada.
- `isainfo -kv`: verifica que la arquitectura sea compatible (64-bit).

Acciones requeridas según comprobaciones:

- Añadir módulos de RAM si no cumple el mínimo establecido.
- Configurar particiones adicionales de swap si espacio es insuficiente.
- Borrar archivos en `/tmp` o cambiar variable de entorno `TMPDIR` si falta espacio.
- Verificar compatibilidad de arquitectura (64-bit) antes de continuar.

Para garantizar el funcionamiento óptimo de Oracle, se llevó a cabo una configuración detallada de la red que permitiera la comunicación eficiente entre los nodos del clúster. La infraestructura de red se diseñó cuidadosamente para proporcionar la conectividad necesaria y asegurar la disponibilidad de los servicios de Oracle.

3.3 Topología

Se ha implementado y simulado una topología de red para facilitar la conectividad entre dos edificios corporativos, A y B, separados por una considerable distancia y conectados a través de Internet mediante un proveedor de servicios (ISP). Esta topología, meticulosamente diseñada, está estructurada para asegurar una comunicación eficiente y segura entre los distintos pisos de cada edificio.

Se ofrece un detalle exhaustivo de la configuración de los routers, destacando la implementación del protocolo OSPF para el intercambio de información de enrutamiento entre los dispositivos de red. También se describe en detalle la configuración de listas de acceso, las cuales refuerzan la seguridad de la red al permitir un control más preciso sobre el tráfico de datos.

Además, se aborda la configuración de seguridad, que incluye la implementación de contraseñas y restricciones de acceso para salvaguardar la integridad y confidencialidad de la red corporativa. A través de imágenes detalladas y configuraciones finales, este capítulo proporciona una visión completa y comprensible de la topología de red implementada, resaltando su robustez y eficacia en el entorno empresarial.

En la Figura 3.1 se presenta la implementación y simulación de la red que conecta dos pisos de una empresa separados por una considerable distancia a través de Internet (ISP). Incluye los edificios A y B, cada uno con 3 pisos, un router en cada piso y un switch que facilita la interconexión de dispositivos en la LAN de cada piso. La conexión entre los routers se realiza mediante el

protocolo OSPF y cables seriales. Además, las Figuras 3.2 y 3.3 ofrecen vistas detalladas de los edificios A y B, respectivamente.

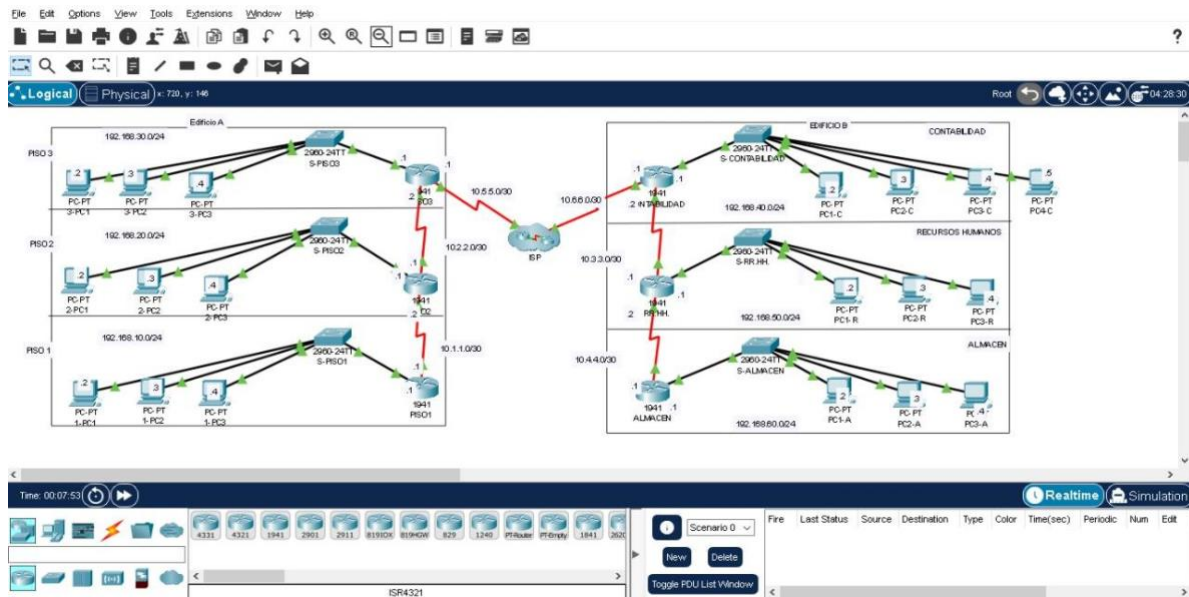


Figura 3.1. Implementación de Topología.

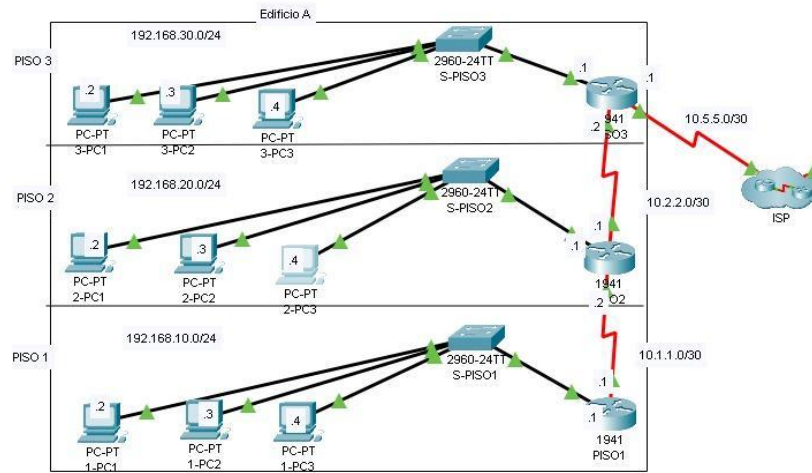


Figura 3.2. Vista detallada Edificio A.

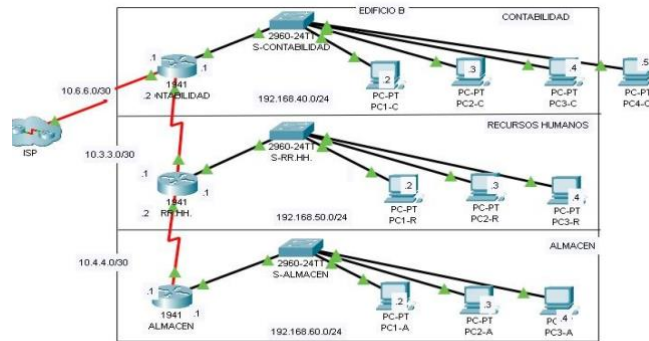


Figura 3.3. Vista detallada Edificio B.

La configuración IPv4 en uno de los hosts asegura una conectividad eficiente en las diversas redes de cada piso. Esta configuración, visible en la figura 3.4, asigna direcciones IP únicas a cada dispositivo, facilitando una comunicación fluida y precisa dentro de la red corporativa. Garantiza la correcta identificación y enrutamiento de datos, contribuyendo así a una infraestructura de red estable y confiable en cada uno de los pisos del edificio.

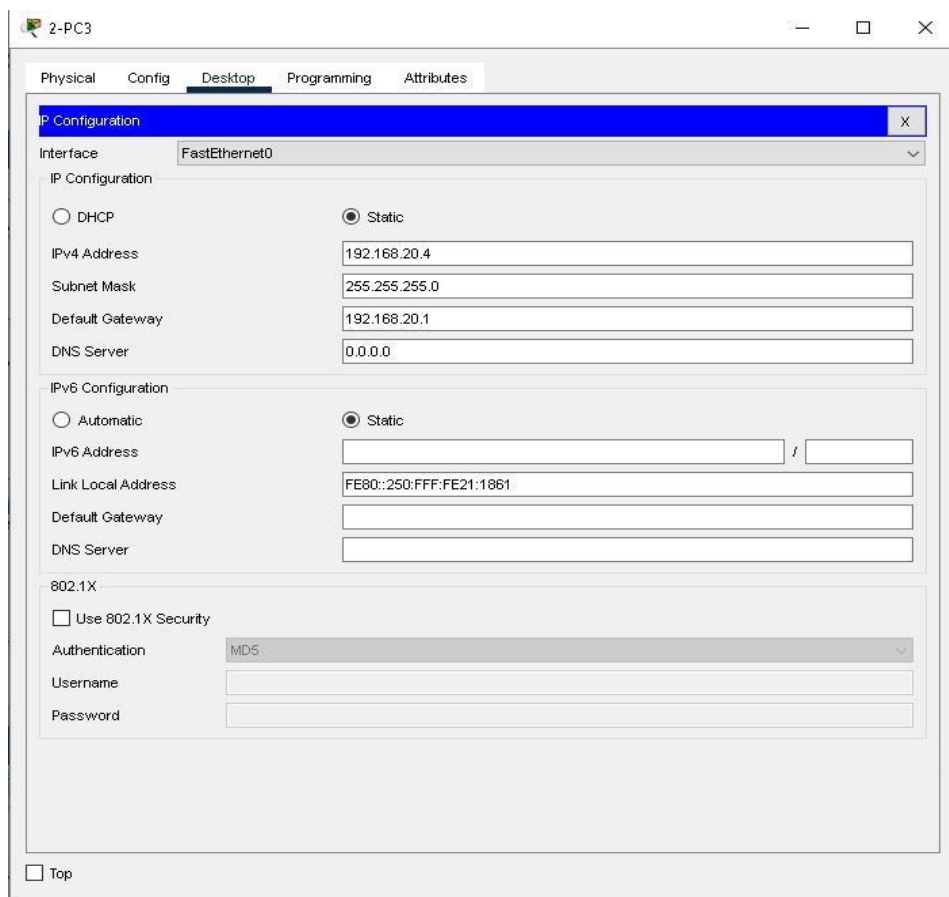
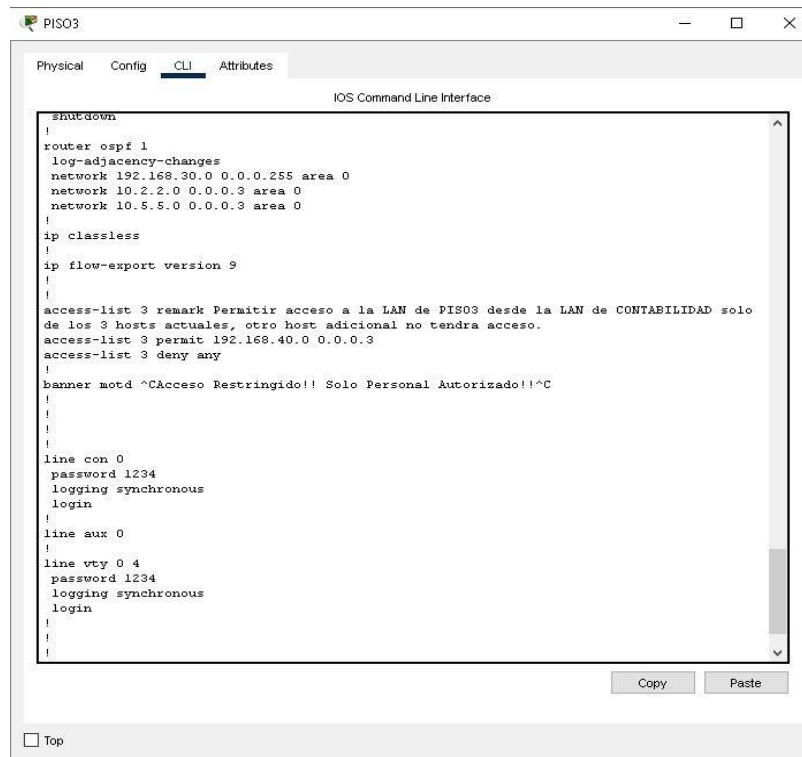


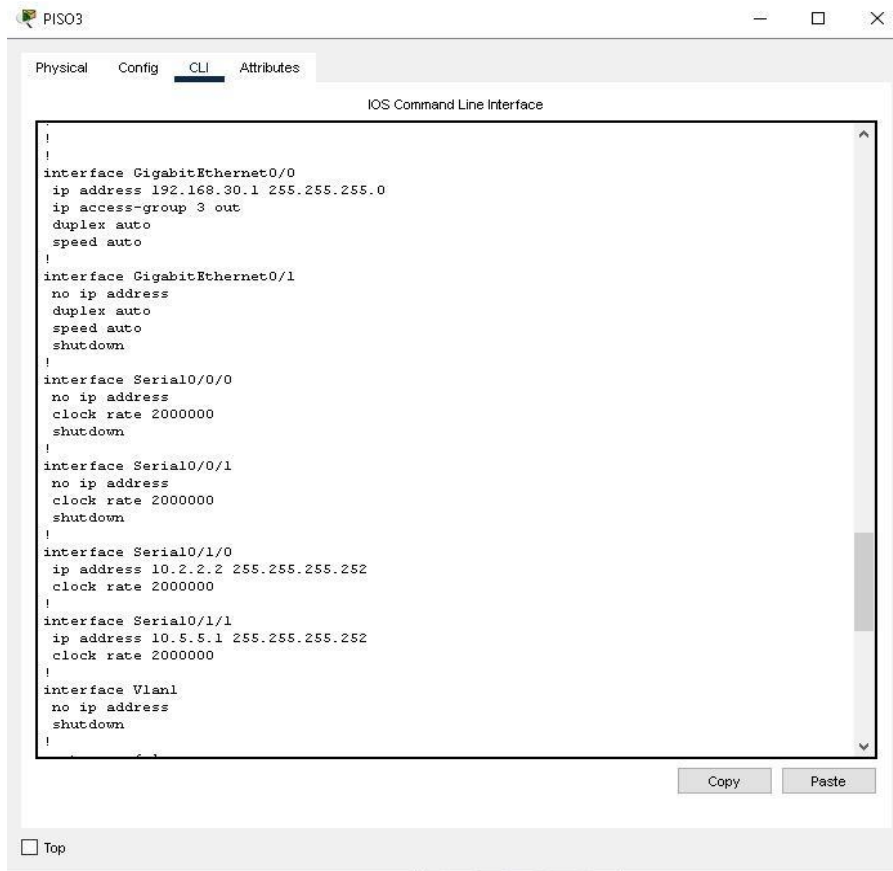
Figura 3.4. Configuración IPv4 en uno de los hosts conectados a las diferentes redes de cada piso.

La configuración final de los routers en cada piso ofrece una visión completa de la infraestructura de red. Esto se puede observar en las figuras 3.5 y 3.6, que muestran, como ejemplo, la configuración del router del PISO3. Esta configuración incluye la asignación de direcciones IP a cada puerto para una identificación precisa de la red. Además, se detalla la configuración del protocolo OSPF, garantizando una eficiente comunicación entre los dispositivos vecinos. Se implementan también listas de acceso numeradas y detalladas para un control preciso del tráfico de datos, junto con medidas de seguridad como contraseñas y accesos restringidos, fortaleciendo así la integridad y la confidencialidad de la red.



```
shutdow
n
router ospf 1
log-adjacency-changes
network 192.168.30.0 0.0.0.255 area 0
network 10.2.2.0 0.0.0.3 area 0
network 10.5.5.0 0.0.0.3 area 0
!
ip classless
!
ip flow-export version 9
!
!
access-list 3 remark Permitir acceso a la LAN de PISO3 desde la LAN de CONTABILIDAD solo
de los 3 hosts actuales, otro host adicional no tendra acceso.
access-list 3 permit 192.168.40.0 0.0.0.3
access-list 3 deny any
!
banner motd ^CAcceso Restringido!! Solo Personal Autorizado!!^C
!
!
!
!
line con 0
password 1234
logging synchronous
login
!
line aux 0
!
line vty 0 4
password 1234
logging synchronous
login
!
!
!
```

Figura 3.5. Configuración Router Piso 3.



The screenshot shows a web-based configuration interface for a router named PISO3. The 'CLI' tab is active, displaying the following configuration commands:

```
!
!
interface GigabitEthernet0/0
 ip address 192.168.30.1 255.255.255.0
 ip access-group 3 out
 duplex auto
 speed auto
!
interface GigabitEthernet0/1
 no ip address
 duplex auto
 speed auto
 shutdown
!
interface Serial0/0/0
 no ip address
 clock rate 2000000
 shutdown
!
interface Serial0/0/1
 no ip address
 clock rate 2000000
 shutdown
!
interface Serial0/1/0
 ip address 10.2.2.2 255.255.255.252
 clock rate 2000000
!
interface Serial0/1/1
 ip address 10.5.5.1 255.255.255.252
 clock rate 2000000
!
interface Vlan1
 no ip address
 shutdown
!
```

At the bottom of the CLI window, there are 'Copy' and 'Paste' buttons, and a 'Top' button with a checkbox.

Figura 3.6. Configuración Router Piso 3.

3.4 Componentes clave de la infraestructura Grid de Oracle

Oracle ASM es un componente de almacenamiento de datos que proporciona una gestión simplificada de archivos de base de datos. ASM crea un pool de almacenamiento abstracto de discos, lo que permite a la base de datos utilizar el almacenamiento de manera más eficiente y resistente a fallos. ASM también equilibra automáticamente la carga de los archivos de la base de datos a través de todos los discos disponibles, lo que puede mejorar el rendimiento general (ver Fig. 7).

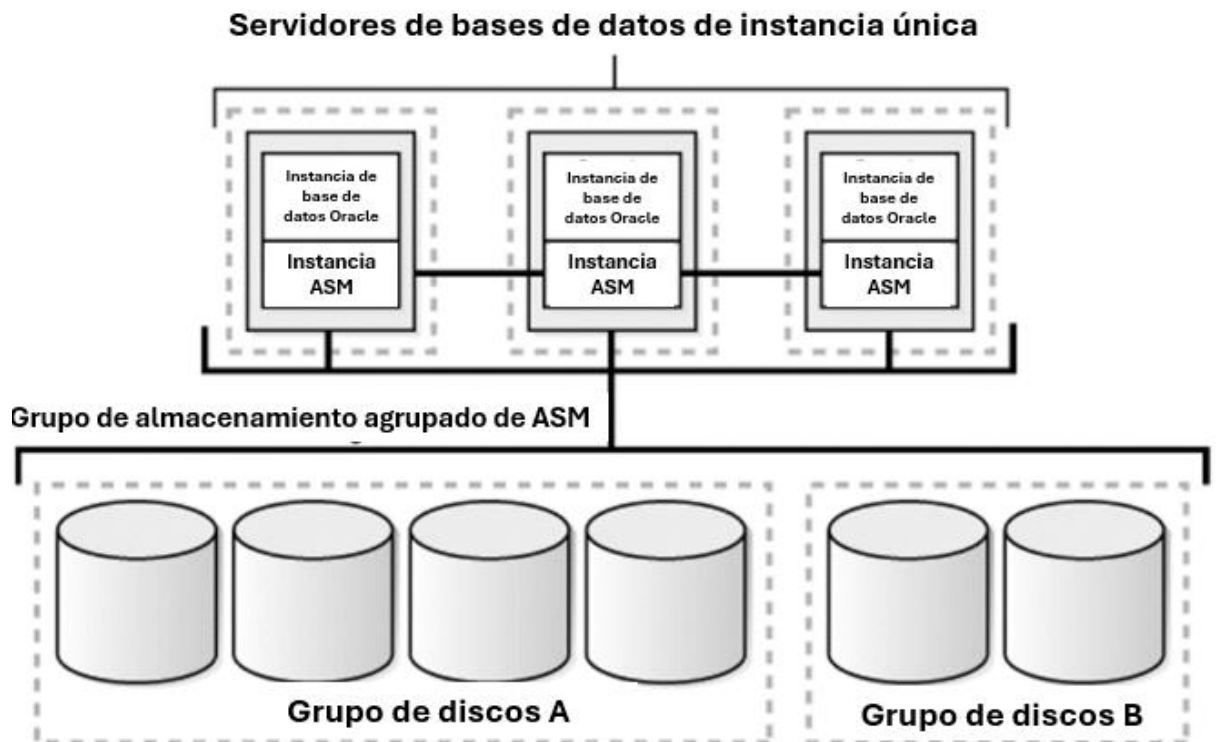


Figura 3.7. Se representan dos grupos de discos, un grupo de disco tiene cuatro discos y el otro tiene dos discos.

Clusterware: Oracle Clusterware es el software de gestión de clústeres de Oracle. Proporciona la infraestructura necesaria para agrupar varios servidores

para que funcionen como un solo sistema. Clusterware es responsable de la detección de fallos y proporciona la infraestructura necesaria para una recuperación automática. También es el fundamento para la Alta Disponibilidad (HA) en productos de Oracle.

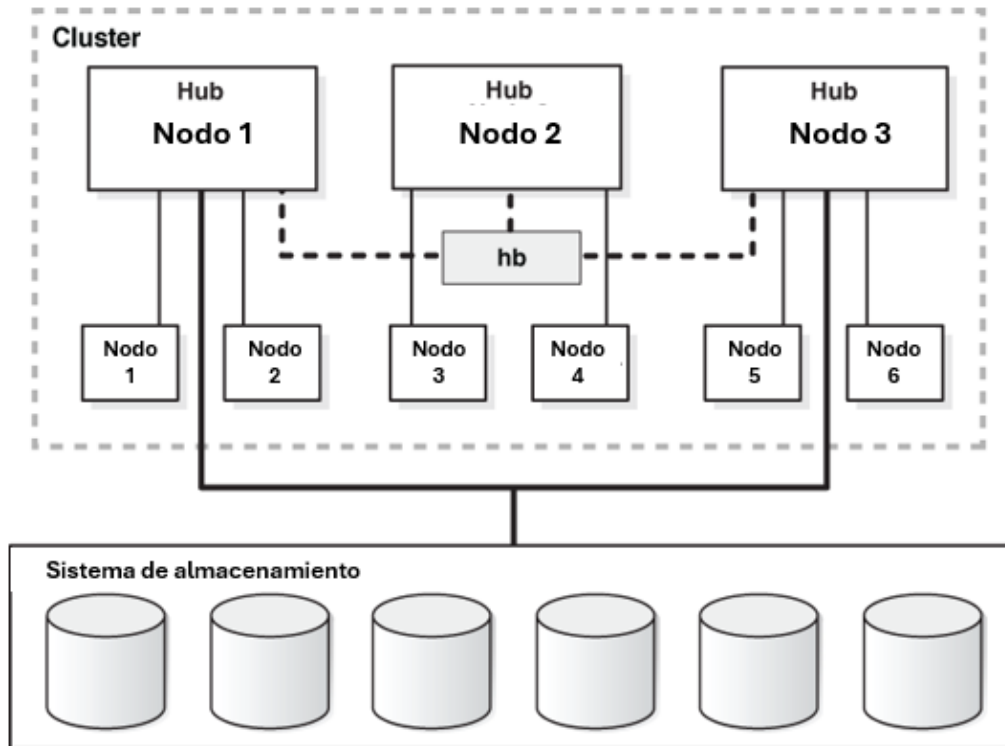


Figura 3.8. Se muestra cómo se organiza el Oracle Clusterware para gestionar bases de datos y aplicaciones en servidores conectados.

En la figura 3.8 se tienen nodos principales llamados "Hub Nodes" con acceso directo al almacenamiento compartido. Otros nodos acceden a través de estos. El Clusterware automatiza la gestión de recursos, asegura la disponibilidad y elimina la necesidad de software de clúster propietario. Sus componentes esenciales son archivos de votación y el Registro de clúster de Oracle (OCR), ambos almacenados en almacenamiento compartido.

Base de Datos (Oracle RAC): Oracle Real Application Clusters (RAC) es una opción para la Base de Datos de Oracle que proporciona alta disponibilidad y escalabilidad para la Base de Datos de Oracle. Oracle RAC utiliza Clusterware para coordinar múltiples instancias de la Base de Datos de Oracle que funcionan en diferentes servidores. Las instancias de la base de datos acceden a una única base de datos, lo que permite que el sistema sobreviva a un fallo de servidor y proporcione escalabilidad al agregar más servidores para aumentar el rendimiento del sistema (ver Fig. 3.9).

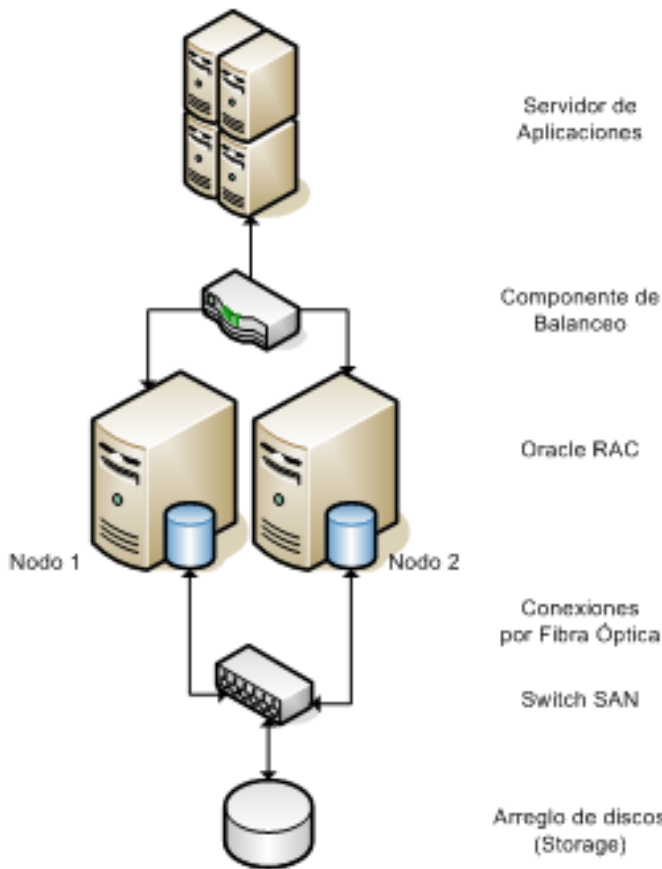


Figura 3.9. Oracle RAC con balanceo de cargas

3.5 Beneficios de usar la infraestructura de Grid

1. Mayor utilización de los recursos informáticos: Mediante la agrupación y el intercambio de recursos, se puede lograr una mayor utilización de los recursos informáticos.
2. Alta disponibilidad: Significa que los sistemas y aplicaciones pueden seguir funcionando incluso en caso de fallos.
3. Rendimiento predecible y extremo: La infraestructura está diseñada para proporcionar un rendimiento extremo y predecible, lo que puede ser crucial para las operaciones de negocio.
4. Escalabilidad elástica: Puede escalar de manera elástica para satisfacer las demandas cambiantes, lo que puede ser especialmente útil en entornos de negocio dinámicos.
5. Gestión centralizada eficaz: Permite una gestión centralizada eficaz, incluyendo el aprovisionamiento de recursos dinámico y automatizado.
6. Procesamiento paralelo y escalabilidad: Permite que múltiples aplicaciones de Oracle Database se ejecuten en múltiples servidores y proporcionen procesamiento paralelo y escalabilidad.
7. Tolerancia a fallos: Si falla una aplicación de base de datos en Grid, cualquier transacción en ejecución es recogida y completada por las copias de trabajo restantes.

Identificación de los requisitos de almacenamiento

1. Planificación de grupos de discos:
 - Determine los requisitos de grupos de discos según la configuración de clúster deseada.
 - Los clústeres de Oracle Domain Services requieren dos grupos de discos independientes para almacenar archivos de Oracle Clusterware y el repositorio GIMR.

2. Decisión de uso de Oracle ASM:
 - Decida si utilizará Oracle ASM para archivos de Oracle Database, archivos de recuperación y binarios de Oracle Database.
 - Se pueden usar sistemas de archivos compartidos para un tipo de archivo y Oracle ASM para otro.

3. Tipos de archivos de Oracle Clusterware:
 - Pueden almacenarse en Oracle ASM o en sistemas de archivos compartidos.
 - Los archivos de votación deben almacenarse en Oracle ASM en clústeres de Oracle Domain Services.

4. Niveles de redundancia de Oracle ASM:
 - Alta redundancia ofrece la máxima fiabilidad, pero a un mayor costo.
 - La redundancia normal es adecuada si no se cuenta con protección independiente contra la pérdida de datos.
 - La redundancia externa depende de dispositivos de almacenamiento con sus propios mecanismos de protección.
 - La redundancia flexible permite cambiar la redundancia de archivos según las necesidades.

5. Cálculo de espacio en disco:
 - La fórmula para calcular los requisitos de espacio en disco considera varios factores, incluidos los tipos de redundancia y el tamaño de unidad de asignación.
 - Puede utilizar la siguiente fórmula para calcular los requisitos de espacio en disco (en MB) para los archivos de OCR y votación, así como los metadatos de Oracle ASM:

$$\text{total} = [2 * \text{ausize} * \text{disks}] + [\text{redundancy} * (\text{ausize} * (\text{all_client_instances} + \text{nodes} + \text{disks} + 32) + (64 * \text{nodes}) + \text{clients} + 543)]$$

redundancy = Número de espejos: externos = 1, normales = 2, altos = 3, flexibles = 3.

ausize = Tamaño AU de metadatos en megabytes

all_client_instance = Suma de todos los clientes de base de datos e instancias de proxy ACFS

nodes = Número de nodos en el clúster.

clients - Número de instancias de base de datos para cada nodo.

disks: número de discos en el grupo de discos.

6. Identificación de grupos de errores:

- Pueden personalizarse para mejorar la tolerancia a fallos.
- Se deben definir antes o después de la instalación según las necesidades y mediante herramientas específicas.

7. Identificación de dispositivos de disco adecuados:

- Los dispositivos deben tener el mismo tamaño y características de rendimiento.
- Evite especificar varias particiones en un solo disco físico como dispositivo de grupo de discos.
- No se recomienda utilizar volúmenes lógicos como dispositivos en grupos de discos de Oracle ASM, salvo que sea estrictamente necesario.

Tabla 8. Espacio de almacenamiento total disponible de Oracle Clusterware requerido por el tipo de implementación de Oracle Cluster.

Nivel de redundancia	Grupo de discos DATA	Grupo de discos MGMT	Aplicación de parches y aprovisionamiento de Oracle Fleet
Externo	1 GB	28 GB Cada nodo más allá de cuatro: 5 GB	1 GB
Normal	2 GB	56 GB Cada nodo	2 GB

		más allá de cuatro: 5 GB	
Alto/Flexible/Extendido	3 GB	84 GB Cada nodo más allá de cuatro: 5 GB	3 GB

Oracle sugiere emplear un grupo de discos independiente para los archivos de respaldo de GIMR y Oracle Clusterware, separado de los datos. Además, al agregar nuevos nodos al clúster, es necesario incrementar el espacio de almacenamiento en el grupo de discos que contiene estos archivos (ver tabla 9).

Por defecto, los nuevos despliegues de Oracle Standalone Cluster se configuran con Oracle Fleet Patching and Provisioning para aplicar parches específicamente a ese clúster. Esta configuración implica tener un sistema de archivos ACFS mínimo, que se establece automáticamente en el mismo grupo de discos que GIMR (ver tabla 10).

Tabla 9 Requisitos mínimos de espacio disponible para el clúster independiente de Oracle sin configuración de GIMR.

Niveles de redundancia	Grupo de discos DATA	Aplicación de parches y aprovisionamiento de Oracle	Almacenamiento total
------------------------	----------------------	---	----------------------

		Fleet	
Externo	1 GB	1 GB	2 GB
Normal	2 GB	2 GB	4 GB
Alto/Flexible/Ext endido	3 GB	3 GB	6 GB

Oracle recomienda emplear un grupo de discos independiente para los archivos de respaldo de Oracle Clusterware, separado del grupo de discos DATA. El tamaño inicial del clúster independiente de Oracle abarca hasta cuatro nodos. Para cada nuevo nodo agregado al clúster, es necesario agregar espacio de almacenamiento adicional al grupo de discos que contiene los archivos de respaldo de Oracle Clusterware. Por defecto, todos los nuevos despliegues de Oracle Standalone Cluster se configuran con Oracle Fleet Patching and Provisioning exclusivamente para aplicar parches a ese clúster. Esta configuración implica tener un sistema de archivos ACFS mínimo, el cual se establece automáticamente.

Por defecto, la asignación inicial de espacio para el GIMR está diseñada para el clúster de Oracle Domain Services y hasta cuatro clústeres de miembros de Oracle. Para cada clúster de miembros de Oracle más allá de cuatro, es necesario agregar espacio de almacenamiento adicional. Durante la instalación, se realizan evaluaciones de los requisitos de espacio de almacenamiento de TFA para asegurar que pueda crecer hasta su tamaño máximo. Solo se asigna el espacio mínimo para el sistema de archivos ACFS, que se extiende automáticamente hasta su valor máximo, según sea necesario.

Oracle sugiere asignar espacio de almacenamiento anticipadamente para la configuración más grande prevista del clúster de Oracle Domain Services, siguiendo las directrices proporcionadas en la tabla correspondiente.

Opciones de redundancia para Clusterware:

Normal: Implica un solo nodo con una copia de los recursos de Clusterware. En este caso, si el nodo falla, los recursos de Clusterware no estarán disponibles hasta que el nodo se recupere.

Alta: Esta opción proporciona una mayor resistencia a fallos al mantener una copia de los recursos de Clusterware en dos nodos. Si un nodo falla, el otro nodo, que tiene una copia de los recursos de Clusterware, puede continuar ejecutando las aplicaciones. Esto garantiza que las aplicaciones puedan seguir funcionando incluso si uno de los nodos falla.

Externa: Un solo nodo con una copia de los recursos de Clusterware en un dispositivo de almacenamiento externo. Si el nodo falla, el dispositivo de almacenamiento externo puede usarse para restaurar los recursos de Clusterware. Esto proporciona una capa adicional de protección contra fallos, ya que los recursos de Clusterware pueden restaurarse incluso si el nodo no está disponible.

Oracle ASM de una sola instancia con grupos de discos independientes:
En este diagrama, la base de datos Oracle de una sola instancia, DB2, que utiliza exclusivamente el grupo de discos de Oracle ASM DG2, se

ejecuta en Node2 después de una migración tras error del grupo de recursos DB2-rg a Node2. En la parte superior del diagrama se muestran las relaciones de las instancias de la base de datos Oracle con las instancias de Oracle ASM únicas en Node1 y Node2. La opción Oracle ASM de una sola instancia de Node2 ahora proporciona servicio al grupo de discos de Oracle ASM DG2, mientras que Oracle ASM de una sola instancia de Node1 proporciona servicio al grupo de discos de Oracle ASM DG1.

En la parte inferior del diagrama aparecen representados los grupos de recursos existentes de Oracle Solaris Cluster y los recursos para las bases de datos Oracle de una sola instancia, así como sus requisitos para los servicios de la opción Oracle ASM de una sola instancia.

Creación de Grupos de Discos

En la figura 11 se muestra cómo se utiliza la herramienta ASMCA (ASM Configuration Assistant) para crear grupos de discos. Especifica los discos que formarán parte de cada grupo.

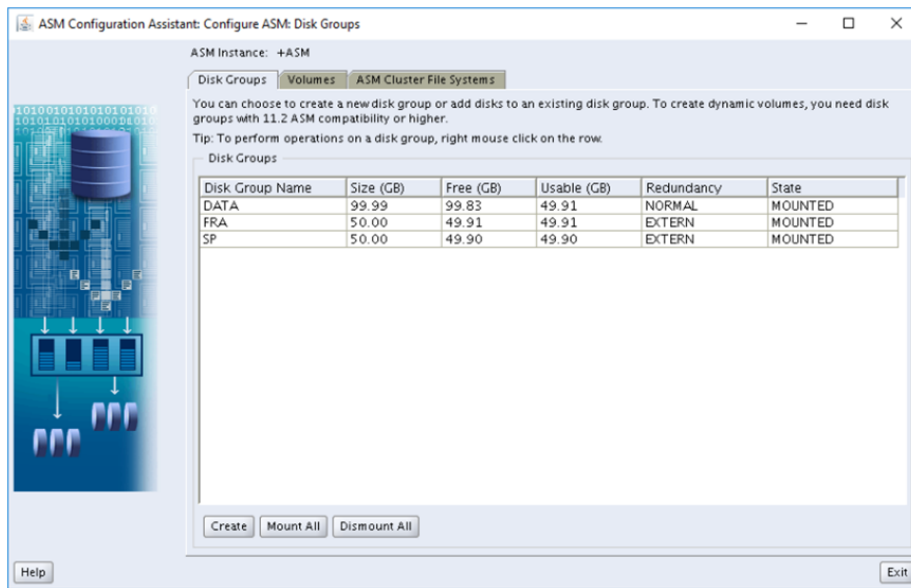


Figura 3.10. Interfaz del Asistente de Configuración ASM mostrando la configuración de grupos de discos ASM.

Se observan en la figura 1.10 las representaciones gráficas de unidades de disco en el lado izquierdo, con diferentes tonos de azul y morado indicando varios discos o unidades de almacenamiento.

Parámetros de configuración

Define parámetros como el nombre del grupo, el tamaño de extensión, el número de discos de rebalanceo, entre otros, según las necesidades y la capacidad del clúster.

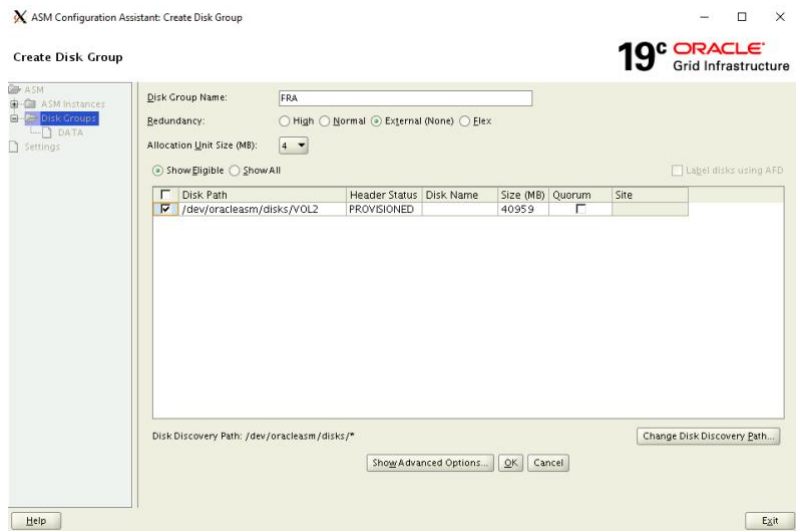


Figura 3.11. Interfaz del Asistente de Configuración ASM: Crear Grupo de Discos de Oracle. Muestra varios campos y opciones disponibles.

Validación y confirmación

Realiza una validación para asegurarte de que los discos estén listos y sean adecuados para ASM. Confirma la creación de los grupos de discos.

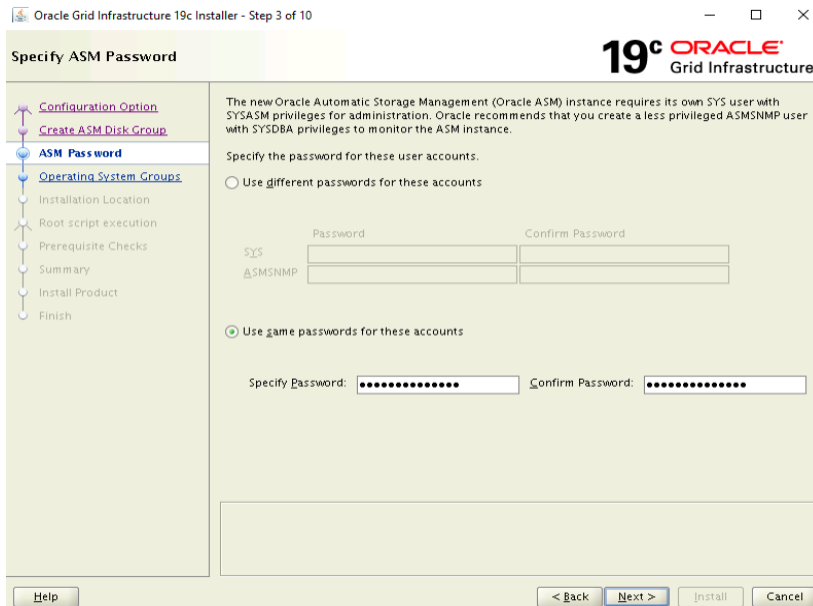


Figura 3.12. Interfaz del instalador de Oracle Grid Infrastructure.

Monitoreo continuo

Después de la creación, monitorea continuamente el rendimiento y el estado de los grupos de discos para asegurar un funcionamiento óptimo.

- Realiza tareas de mantenimiento periódicas, como la adición de nuevos discos o la eliminación de discos defectuosos.
- Asegúrate de que los grupos de discos tengan suficiente espacio libre para acomodar el crecimiento de la base de datos.

Tanto el mirroring como el stripping son técnicas utilizadas para mejorar la eficiencia y la redundancia del almacenamiento:

- **Mirroring (Espejado):**

El mirroring se refiere a la creación de copias duplicadas de los datos en diferentes discos.

En ASM, esto se logra mediante la creación de discos de espejo (mirror disks) dentro de un grupo de discos. Cada disco de espejo contiene una copia idéntica de los datos almacenados en otro disco. Si un disco falla, ASM automáticamente redirige las operaciones de E/S al disco de espejo, garantizando la disponibilidad continua de los datos.

El mirroring se utiliza para proteger contra fallos de hardware y aumentar la confiabilidad.

- **Stripping (Entrelazado):**

El striping implica dividir los datos en segmentos más pequeños y distribuirlos en varios discos.

Cada segmento se almacena en un disco diferente, lo que permite una mejor utilización del ancho de banda de E/S.

El striping mejora el rendimiento al permitir que múltiples discos trabajen simultáneamente para leer o escribir datos. No proporciona redundancia por sí mismo, pero se combina comúnmente con el mirroring para lograr un equilibrio entre rendimiento y protección.

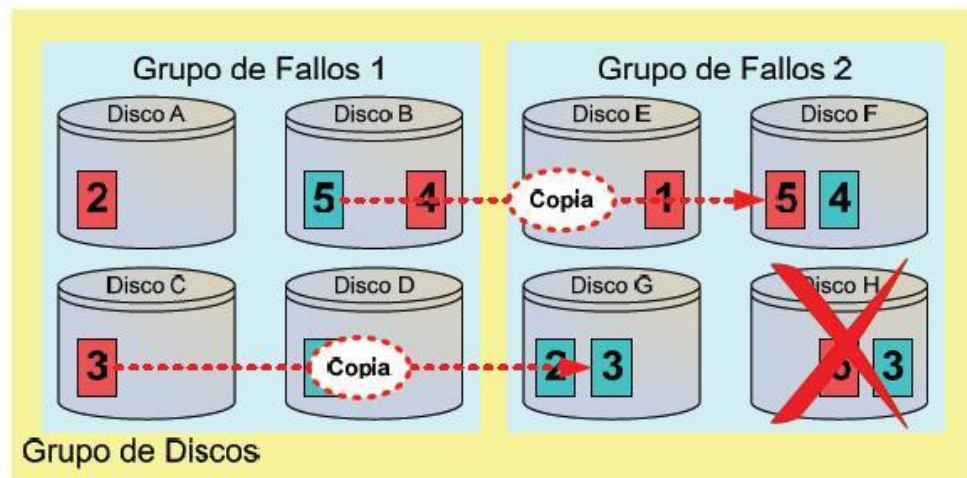


Figura 3.13. Mirroring y Striping automático de ASM.

Como se muestra en la figura 3.13 para utilizar un grupo de discos de Oracle ASM existente, siga estos pasos:

1. Utilice Oracle Enterprise Manager Cloud Control o la herramienta de línea de comandos Oracle ASM (asmcmd) para verificar la existencia de grupos de discos y asegurarse de que haya suficiente espacio disponible en el grupo de discos.

2. Conéctese a la instancia de Oracle ASM y, si es necesario, iníciela utilizando los siguientes comandos:

```
$ORACLE_HOME/bin/asmcmd
```

```
ASMCMD> startup
```

3. Utilice el comando "lsdg" para mostrar información sobre los grupos de discos montados. Puede usar los siguientes comandos:

```
ASMCMD> lsdg o $ORACLE_HOME/bin/asmcmd -p lsdg
```

4. En la salida, identifique un grupo de discos con el nivel de redundancia adecuado y tome nota del espacio libre disponible.
5. Si es necesario, instale o identifique los dispositivos de disco adicionales requeridos para satisfacer los requisitos de almacenamiento de su instalación.

Al especificar la cadena de detección de discos de Oracle ASM, se establece el parámetro de inicialización ASM_DISKSTRING. Este parámetro indica a Oracle ASM las rutas donde buscar discos al inicializar una instancia.

La sintaxis de la cadena de detección varía según el sistema operativo, las bibliotecas ASMLib y la presencia de discos Oracle Exadata. Se pueden usar los nombres de ruta de acceso del sistema operativo como parte de la cadena de detección.

El valor predeterminado del parámetro puede no detectar todos los discos en algunas situaciones. Además, si se emplea software de múltiples rutas, es posible que genere pseudo dispositivos en rutas diferentes a las predeterminadas del sistema operativo. En estos casos, se requiere ajustar el parámetro `ASM_DISKSTRING` para abarcar todas las rutas relevantes.

Selección de discos para utilizar con grupos de discos de Oracle ASM

- Si no existe un grupo de discos adecuado en el sistema, se debe instalar o identificar los dispositivos de disco adecuados para agregarlos a un nuevo grupo de discos. Se deben utilizar las siguientes directrices para identificar los dispositivos de disco adecuados:
- Todos los dispositivos de un grupo de discos de Oracle ASM deben tener el mismo tamaño y las mismas características de rendimiento.
- No especifique varias particiones en un solo disco físico como dispositivo de grupo de discos. Oracle ASM espera que cada dispositivo de grupo de discos esté en un disco físico independiente.
- Las particiones lógicas no compartidas no están soportadas por Oracle RAC. Para utilizar particiones lógicas para la base de datos Oracle RAC, debe utilizar volúmenes lógicos compartidos creados por un gestor de volúmenes lógicos como `.fdisk`

- Aunque puede especificar un volumen lógico como dispositivo en un grupo de discos de Oracle ASM, Oracle no recomienda su uso porque agrega una capa de complejidad que no es necesaria con Oracle ASM. Además, Oracle RAC requiere un gestor de volúmenes lógicos de clúster en caso de que decida utilizar un volumen lógico con Oracle ASM y Oracle RAC.

Cadena de Descubrimiento de Discos ASM

- Inicio de ASM Instances: Inicia la instancia ASM utilizando el servidor Oracle Grid Infrastructure
- Exploración Automática: ASM realiza una exploración automática de los discos en busca de aquellos que se pueden agregar al grupo de discos ASM.
- Identificación de Discos Candidatos: ASM identifica discos candidatos en función de su estado y disponibilidad. Estos discos pueden ser añadidos al grupo de discos ASM.
- Validación y Evaluación: ASM valida la idoneidad de los discos candidatos mediante criterios como capacidad, rendimiento y estado de salud.
- Selección y Adición: Selecciona los discos que cumplen con los requisitos y los añade al grupo de discos ASM. Este proceso se realiza de manera dinámica y automática.

- **Rebalanceo Automático:** ASM realiza un rebalanceo automático para distribuir uniformemente los datos entre los discos y garantizar un rendimiento equitativo.
- **Notificación de Eventos:** En caso de cualquier cambio en la configuración o eventos inesperados, ASM notifica para una rápida intervención.

Creación de archivos en un dispositivo NAS para su uso con Oracle Automatic Storage Management

Para utilizar archivos en un dispositivo NAS con Oracle Automatic Storage Management (ASM), sigue estos pasos:

1. Crea un directorio exportado para los archivos del grupo de discos en el dispositivo NAS.
2. Cambia al usuario root.
3. Crea un directorio de punto de montaje en el sistema local, por ejemplo: /mnt/oracleasm.
4. Añade una entrada al archivo de montaje /etc/fstab para asegurar que el sistema monte el sistema de archivos NFS al reiniciar.
5. Monta el NFS en el sistema local con el comando mount /mnt/oracleasm.

6. Crea un nombre para el grupo de discos debajo del punto de montaje, por ejemplo, /mnt/oracleasm/sales1.
7. Crea archivos zero-padded en este directorio con el comando dd, por ejemplo, `dd if=/dev/zero of=/mnt/oracleasm/sales1/disk1 bs=1024k count=1000`.
8. Ajusta los permisos y propietarios de los archivos y directorios creados con `chown` y `chmod`.
9. Durante la instalación de Oracle Database, edita la cadena de descubrimiento de discos de Oracle ASM para que coincida con los nombres de archivos que creaste, por ejemplo: /mnt/oracleasm/sales1/.

Es importante tener en cuenta que crear múltiples archivos zero-padded en el mismo dispositivo NAS no protege contra fallos del NAS. Se recomienda crear un archivo por cada dispositivo NAS y espejearlos utilizando la tecnología de Oracle ASM

4 Resultados

Una red de computadoras es cualquier grupo de dispositivos de cómputo interconectados que pueden enviar o recibir datos. Un dispositivo de cómputo no es sólo una computadora, es cualquier dispositivo que puede correr un programa, como una tableta, teléfono o sensor inteligente.

En este proyecto de tesis comenzamos por construir la red con dispositivos o terminales:



Figura 4.1. Red de computadoras interconectadas.

Ahora hay complejidad adicional. ¿Cómo puede cada dispositivo saber si los datos que le llegan están destinados para ellos o para su vecino? Esta red simple necesitará un esquema de direccionamiento como se ve en la Figura 4.1.

Para exportar o respaldar el contenido de una de tus bases de datos mediante PHPMyAdmin sigue estos sencillos pasos:

4.1 Parches, Drivers y Dns

La gestión de parches es el proceso de aplicar actualizaciones emitidas por el proveedor para cerrar las vulnerabilidades de seguridad y optimizar el rendimiento del software y los dispositivos. A veces se considera que la gestión de parches es parte de la gestión de vulnerabilidades.

En la práctica, la gestión de parches consiste en equilibrar la ciberseguridad con las necesidades operativas de la empresa. Los hackers pueden explotar las vulnerabilidades en el entorno de TI de una empresa para lanzar ciberataques y propagar malware. Los proveedores publican actualizaciones, denominadas "parches" para corregir estas vulnerabilidades. Sin embargo, el proceso de aplicación de parches puede interrumpir los flujos de trabajo y crear tiempos de inactividad para la empresa. La gestión de parches tiene como objetivo minimizar ese tiempo de inactividad al optimizar la implementación de parches.

Estos parches pueden mejorar la seguridad, aumentar el rendimiento y estimular la productividad.

Los parches de seguridad abordan riesgos de seguridad específicos, a menudo al corregir una vulnerabilidad en particular.

Con frecuencia, los hackers se dirigen a los activos que no tienen parches, por lo que, si no se aplican las actualizaciones de seguridad, la empresa puede quedar expuesta a violaciones de seguridad. Por ejemplo, el ransomware WannaCry 2017 se propaga a través de una vulnerabilidad de Microsoft Windows para la que se había emitido un parche. Los delincuentes cibernéticos atacaron redes en las que los administradores habían omitido la aplicación del parche, infectando más de 200 000 equipos en 150 países.

4.2 Minimizar el tiempo de inactividad

A la mayoría de las empresas les resulta poco práctico descargar y aplicar cada parche para cada activo inmediatamente después de su lanzamiento. Esto se debe a que la aplicación de parches requiere tiempo de inactividad. Los usuarios deben dejar de trabajar, cerrar sesión y reiniciar los sistemas principales para aplicar los parches.

Un proceso formal de gestión de parches permite a las organizaciones priorizar las actualizaciones críticas. La empresa puede obtener los beneficios de estos parches con una interrupción mínima en los flujos de trabajo de los empleados.

Según normas como el Reglamento general de protección de datos (RGPD), la Ley de portabilidad y responsabilidad del seguro médico (HIPAA) y la Norma de seguridad de datos de la industria de tarjetas de pago (PCI-DSS), las empresas deben seguir ciertas prácticas de ciberseguridad. La gestión de parches puede ayudar a las organizaciones a asegurar que los sistemas críticos mantengan el cumplimiento de estas normas [15, 16].

La estandarización también puede evitar que los empleados utilicen aplicaciones y dispositivos peligrosos, obsoletos o incompatibles.

1. Supervisión de parches. Se realiza un seguimiento del estado de los parches de los activos e identificar los activos a los que no se han aplicado parches.
2. Priorización de parches. Algunos parches son más importantes que otros, en especial cuando se trata de parches de seguridad.
3. Pruebas de parches. En ocasiones, los nuevos parches pueden causar problemas, romper integraciones o no corregir las vulnerabilidades que pretenden solucionar.
4. Implementación de parches. Las ventanas para aplicar parches generalmente están definidas por los momentos en que hay pocos empleados, o ninguno, trabajando (poco tráfico en la red).
5. Documentación de parches. Esta documentación mantiene actualizado el inventario de activos y puede demostrar el cumplimiento de las normas de ciberseguridad en caso de una auditoría.

Al igual que una guía telefónica, el sistema de nombres de dominio (DNS) cierra la brecha entre humanos y computadoras al vincular los nombres de dominio a sus respectivas direcciones de Protocolo de Internet (IP).

En un nombre de dominio el cual tiene una zona DNS que lo vincula a los servidores DNS, por lo que también cubriremos cómo puedes cambiar tu zona DNS. Por último, repasaremos qué es la propagación de DNS y el DNSSEC.

Cuando los usuarios introducen un nombre de dominio en la barra de direcciones de su navegador web, son conducidos al sitio que desean visitar. Sin embargo, esta tarea aparentemente instantánea consta de varios pasos conocidos como DNS Lookup o de proceso resolución de DNS.

Para ilustrar mejor el funcionamiento del DNS, aquí presentamos un ejemplo de cómo es el proceso de resolución.

Quieres ir al sitio web de Hostinger, así que introduces el nombre de dominio `hostinger.es` en la barra de direcciones del navegador web. Lo que estás haciendo aquí es enviar una solicitud de DNS.

Luego, tu ordenador comprobará si ya ha almacenado un registro DNS del dominio que has enviado localmente. Un registro DNS es una dirección IP que se corresponde con el nombre del dominio.

En primer lugar, tu ordenador buscará en su archivo `hosts` y en su caché. El archivo `hosts` es un archivo de texto plano que asigna nombres de host a direcciones IP en un sistema operativo, mientras que la caché es los datos temporales almacenados por un componente de hardware o software.

Las direcciones IP correspondientes del servicio DNS suelen encontrarse en la caché del navegador o en la caché del proveedor de servicios de Internet (ISP). Sin embargo, si no se encuentra ninguna dirección IP que coincida en el archivo hosts y en la caché, se añadirán otros pasos adicionales al proceso de resolución.

El Daemon netcd reduce el tiempo que tardan los servicios locales, DNS, NIS y de módulo cargable por el usuario en responder a una consulta almacenando en la memoria caché la respuesta recuperada de los solucionadores.

Cuando el Daemon netcd está en ejecución y configurado para un programa de resolución (por ejemplo, DNS) y una correlación (por ejemplo, hosts), la resolución se realiza primero utilizando las respuestas almacenadas en memoria caché. Si falla, se llama al programa de resolución y el Daemon netcd almacena en memoria caché la respuesta.

El tipo de correlaciones que están soportadas para las resoluciones de módulos locales, NIS y cargables por el usuario son hosts, servicios, redes, protocolos y netgroup. Para DNS, los hosts son el único tipo de correlación que puede utilizar.

Además, para el caso específico de Páginas Amarillas, se han añadido los siguientes mapas:

- passwd.byname
- passwd.byuid

- group.byname
- group.bygid
- netid.byname
- passwd.adjunct.byname

Puede utilizar un archivo de configuración para especificar los programas de resolución y correlaciones que desea configurar. También puede establecer otros parámetros netcd utilizando este archivo. De forma predeterminada, el archivo de configuración utilizado es el archivo `/etc/netcd.conf` . Puede cambiar la vía de acceso de este archivo de configuración utilizando el argumento `-c` del daemon `netcd` . Si el archivo `/etc/netcd.conf` no existe, el daemon `netcd` utiliza los parámetros predeterminados. Puede encontrar un ejemplo de este archivo en el archivo `/usr/samples/tcpip` . No utilice este archivo como archivo de configuración porque se sobrescribirá con una nueva instalación del paquete que contiene el archivo.

También podemos especificar el nivel de depuración utilizando el argumento `-d` . Los niveles de depuración son similares a los utilizados por el daemon `syslogd` . Los mensajes de registro se graban en el archivo `/var/tmp/netcd.log` . Puede alterar temporalmente el valor predeterminado utilizando el archivo de configuración `netcd`. Al igual que con el daemon `syslogd` , puede especificar la rotación para el archivo de registro `netcd`.

Parámetros de netcd

Cuando se inserta una entrada en una memoria caché `netcd`, se le asocia un tiempo de vida (TTL). Puede configurar este TTL utilizando el archivo de

configuración netcd (declaraciones de memoria caché). Para DNS, este TTL es el que contiene la respuesta del DNS.

Para limpiar las memorias caché de las entradas obsoletas, debe ejecutar dos tareas periódicamente, una para limpiar las memorias caché locales y la otra para limpiar las otras memorias caché. Puede establecer la frecuencia de estas tareas utilizando los parámetros `locales_scan_frequency` y `net_scan_frequency` en el archivo de configuración netcd.

Las memorias caché son tablas con hash. El tamaño de las tablas hash se puede controlar utilizando el archivo de configuración netcd y el mandato `netcdctrl` .

Para comunicarse entre las aplicaciones, el daemon netcd utiliza un socket (`/dev/netcd`). Puede configurar el tamaño de la cola de mensajes utilizando el archivo de configuración netcd.

netcd da soporte al controlador de recursos del sistema

El daemon netcd forma parte del grupo de controlador de recursos del sistema (SRC) netcd. A continuación, se muestran los mandatos SRC que puede utilizar para gestionar el daemon netcd:

1. Puede iniciar el daemon netcd utilizando el mandato `startsrc` o detener el daemon netcd utilizando el mandato `stopsrc` .

2. El mandato `lssrc` proporciona una salida de estado breve que incluye el ID de proceso (PID) y el estado del daemon `netcd` .
3. El mandato `lssrc -l` proporciona una salida de estado larga que incluye el PID, el estado del daemon `netcd` , el archivo de configuración utilizado al iniciar el daemon `netcd` y las memorias caché configuradas.

4.3 Certificados SSL

Las empresas y las organizaciones deben agregar certificados SSL a sus sitios web para proteger las transacciones en línea y mantener la privacidad y seguridad de la información del cliente.

En resumen: el certificado SSL mantiene seguras las conexiones a Internet y evita que los delincuentes lean o modifiquen la información transferida entre dos sistemas. Cuando veas un ícono de candado junto a la URL en la barra de direcciones, significa que hay un certificado SSL que protege el sitio web que estás visitando [22].

Desde su creación hace aproximadamente 25 años, ha habido varias versiones del protocolo SSL, las cuales en algún momento se encontraron con problemas de seguridad. Posteriormente, se lanzó una versión renovada y con un nuevo nombre: TLS (Transport Layer Security, Seguridad de capa de transporte), que sigue en uso actualmente. Sin embargo, las iniciales SSL se mantuvieron, por lo que la nueva versión del protocolo se sigue llamando con el nombre antiguo.

Los certificados SSL funcionan garantizando que los datos transferidos entre usuarios y sitios web, o entre dos sistemas, sean imposibles de leer. Utiliza algoritmos de cifrado para cifrar los datos en tránsito, lo que evita que los hackers la información que se envía a través de la conexión. Estos datos incluyen información potencialmente confidencial, como nombres, direcciones, números de tarjetas de crédito u otros detalles financieros.

El proceso funciona de la siguiente manera:

- Un navegador o servidor intenta conectarse a un sitio web (es decir, un servidor web) protegido mediante certificados SSL.
- El navegador o servidor solicita que el servidor web se identifique.
- En respuesta el servidor web envía al navegador o servidor una copia de su certificado SSL.
- El navegador o servidor evalúa si el certificado SSL es confiable. En caso afirmativo, envía una señal al servidor web.
- A continuación, el servidor web devuelve un reconocimiento firmado digitalmente para iniciar una sesión cifrada mediante SSL.
- Los datos cifrados se comparten entre el navegador o servidor y el servidor web.

Este proceso a veces se conoce como “enlace SSL”. Aunque parece ser un proceso largo, se lleva a cabo en milisegundos.

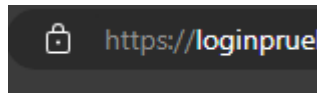


Figura 4.2. Vista del protocolo https en el url del prototipo.

Cuando un sitio web está protegido mediante un certificado SSL, en la URL aparece la sigla HTTPS como se ve en la figura 4.2 (que significa HyperText Transfer Protocol Secure, protocolo de transferencia de hipertexto seguro). Sin un certificado SSL, solo aparecerán las letras HTTP, es decir, sin la S de “seguro”. También se mostrará un ícono de candado en la barra de dirección URL. Esto indica que se trata de un sitio web confiable y brinda tranquilidad a quienes lo visitan [22].

Para ver los detalles de un certificado SSL, puedes hacer clic en el símbolo de candado ubicado en la barra del navegador.

Estos son algunos de los detalles que generalmente se incluyen en los certificados SSL:

1. El nombre de dominio asociado al certificado emitido
2. A qué persona, organización o dispositivo se emitió
3. Qué autoridad de certificación lo emitió
4. La firma digital de la autoridad de certificación
5. Subdominios asociados
6. Fecha de emisión del certificado
7. La fecha de vencimiento del certificado
8. La clave pública (no se revela la clave privada)

Como se menciona en [23], el procesador Intel x86 usa una arquitectura compleja del equipo del conjunto de instrucciones (CISC), lo que significa que

hay un número modesto de registros de propósito especial en lugar de grandes cantidades de registros de uso general. También significa que las instrucciones complejas de propósito especial prevalecerán.

El procesador x86 realiza un seguimiento de su herencia al menos hasta el procesador Intel 8080 de 8 bits. Muchas peculiaridades del conjunto de instrucciones x86 se deben a la compatibilidad con versiones anteriores con ese procesador (y con su variante Zilog Z-80).

Microsoft Win32 usa el procesador x86 en modo plano de 32 bits. Esta documentación se centrará solo en el modo plano.

Registros

La arquitectura x86 consta de los siguientes registros enteros sin privilegios.

Eax	Acumulador
Ebx	Registro base
ecx	Registro de contadores
Edx	Registro de datos: se puede usar para el acceso al puerto de E/S y las funciones aritméticas.
Esi	Registro de índice de origen
Edi	Registro de índice de destino
Ebp	Registro de puntero base
Esp	Puntero de pila

La arquitectura x86 tiene varias convenciones de llamada diferentes. Afortunadamente, todos siguen las mismas reglas de conservación de registros y devolución de función:

- Las funciones deben conservar todos los registros, excepto eax, ecx y edx, que se pueden cambiar a través de una llamada de función, y esp, que se deben actualizar según la convención de llamada.
- El registro eax recibe valores devueltos de función si el resultado es de 32 bits o más pequeños. Si el resultado es de 64 bits, el resultado se almacena en el par edx:eax .

A continuación, se muestra una lista de las convenciones de llamada usadas en la arquitectura x86:

- Win32 (`__stdcall`)

Los parámetros de función se pasan en la pila, se insertan de derecha a izquierda y el destinatario limpia la pila.

- Llamada de método nativa de C++ (también conocida como `thiscall`)

Los parámetros de función se pasan en la pila, se insertan de derecha a izquierda, el puntero "this" se pasa en el registro ecx y el destinatario limpia la pila.

- COM (`__stdcall` para llamadas de método de C++)

Los parámetros de función se pasan en la pila, se insertan de derecha a izquierda, luego se inserta el puntero "this" en la pila y, a continuación, se llama a la función. El destinatario de la llamada limpia la pila.

- `__fastcall`

Los dos primeros argumentos `DWORD` o más pequeños se pasan en los registros ecx y edx . Los parámetros restantes se pasan en la pila, insertados de derecha a izquierda. El destinatario de la llamada limpia la pila.

- `__cdecl`

Los parámetros de función se pasan en la pila, se insertan de derecha a izquierda y el autor de la llamada limpia la pila. La convención de llamada `__cdecl` se usa para todas las funciones con parámetros de longitud variable.

Este es un ejemplo de visualización del registro del depurador:

<code>eax=00000000 ebx=008b6f00 ecx=01010101 edx=ffffffff esi=00000000 edi=00465000</code>
<code>eip=77f9d022 esp=05cffc48 ebp=05cffc54 iopl=0 nv up ei ng nz na po nc</code>
<code>cs=001b ss=0023 ds=0023 es=0023 fs=0038 gs=0000 efl=00000286</code>

4.4 Sincronización (sistemas paralelos)

La sincronización es el proceso de garantizar que los procesos paralelos y distribuidos acuerden un estado coherente de datos, eventos o recursos compartidos. La sincronización es esencial para la corrección, el rendimiento y la fiabilidad de los sistemas paralelos y distribuidos. Por ejemplo, la sincronización puede evitar daños en los datos, interbloqueos, condiciones de carrera o incoherencias entre réplicas [24]. La sincronización es difícil porque los procesos paralelos y distribuidos pueden tener diferentes vistas del estado del sistema, debido a factores como la latencia, los errores de comunicación, la simultaneidad o la falta de determinismo. Además, la sincronización puede incurrir en sobrecarga, complejidad o compensaciones en términos de disponibilidad, coherencia o escalabilidad. Por lo tanto, la sincronización requiere un diseño, análisis e implementación cuidadosos de algoritmos y protocolos que puedan hacer frente a estos desafíos.

Se pueden utilizar diferentes técnicas para implementar la sincronización, según el modelo de sincronización, la arquitectura del sistema y los requisitos de la aplicación. Las técnicas comunes incluyen barreras, que son puntos de sincronización que requieren que todos los procesos alcancen antes de continuar; bloqueos, que permiten que solo un proceso acceda a un recurso compartido a la vez; operaciones atómicas, que realizan una operación de lectura-modificación-escritura en un paso indivisible; señales, que permiten que un proceso notifique a otro proceso de un evento o condición; y las difusiones, que permiten que un proceso envíe un mensaje a todos o algunos procesos de un sistema. Estas técnicas se pueden utilizar para coordinar fases, garantizar la exclusión mutua, implementar la comunicación entre procesos y más.

En esta sección se explica el procedimiento que se utilizó para identificar el tipo de corrupción de datos y, si es factible, cómo reparar los datos.

1. Identificación del tipo de deterioro de datos
2. Reparación de un archivo o directorio dañado
3. Reparación de daños en las agrupaciones de almacenamiento de ZFS

Para reducir al mínimo las posibilidades de que los datos sufran daños, ZFS utiliza sumas de comprobación, redundancia y datos que se reparan a sí mismos. Ahora bien, los datos se pueden dañar si una agrupación no es redundante, cuando una agrupación está en estado "degraded" o si se combina una improbable serie de eventos para dañar varias copias de determinados datos. Sea cual sea el origen, el resultado es el mismo: los datos quedan

dañados y no se puede acceder a ellos. Las medidas requeridas dependen del tipo de datos dañados y su valor relativo. Se pueden dañar dos tipos básicos de datos:

1. Metadatos de agrupación: para abrir una agrupación y acceder a conjuntos de datos, ZFS debe analizar cierta cantidad de datos. Si se dañan estos datos, quedará inaccesible toda la agrupación o partes de la jerarquía de los conjuntos de datos.
2. Datos de objeto: en este caso, el daño afecta a un determinado archivo o directorio. Ello puede hacer que no sea posible acceder a una parte del archivo o directorio, o causar la interrupción del objeto.

Los datos se verifican durante el funcionamiento normal y durante el proceso de limpieza. Para obtener más información sobre cómo verificar la integridad de datos de agrupaciones, consulte Comprobación de integridad de sistema de archivos ZFS.

Identificación del tipo de deterioro de datos

De forma predeterminada, el comando `zpool estatus` avisa únicamente de la presencia de daños, pero no indica su ubicación como en la figura 4.5. Por ejemplo:

```

# zpool status monkey
pool: monkey
state: ONLINE
status: One or more devices has experienced an error resulting in data
corruption. Applications may be affected.
action: Restore the file in question if possible. Otherwise restore the
entire pool from backup.
see: http://www.sun.com/msg/ZFS-8000-8A
scrub: scrub completed after 0h0m with 8 errors on Tue Jul 13 13:17:32 2010
config:

    NAME      STATE    READ WRITE CKSUM
    monkey    ONLINE   8     0     0
    c1t1d0    ONLINE   2     0     0
    c2t5d0    ONLINE   6     0     0

errors: 8 data errors, use '-v' for a list

```

Figura 4.5. comando zpool para verificación de daños.

Si un archivo o directorio resultasen dañados, según el tipo de corrupción, el sistema podría seguir funcionando. Si en el sistema no hay copias de los datos de buena calidad, cualquier daño que tenga lugar será irreparable. Si los datos son importantes, la única alternativa es recuperarlos a partir de una copia de seguridad. Aun así, debe poder realizar una recuperación sin necesidad de restaurar toda la agrupación.

Si se ha dañado un bloque de datos de archivo, el archivo se puede eliminar sin problemas; de este modo, el error desaparece del sistema. Utilice el comando `zpool status -v` para ver en pantalla una lista con nombres de archivos que tienen errores constantes como se ve en la figura 4.6. Por ejemplo:

```

# zpool status -v
pool: monkey
state: ONLINE
status: One or more devices has experienced an error resulting in data
corruption. Applications may be affected.
action: Restore the file in question if possible. Otherwise restore the
entire pool from backup.
see: http://www.sun.com/msg/ZFS-8000-8A
scrub: scrub completed after 0h0m with 8 errors on Tue Jul 13 13:17:32 2010
config:

        NAME      STATE      READ WRITE CKSUM
monkey   ONLINE    8     0     0
  c1t1d0  ONLINE    2     0     0
  c2t5d0  ONLINE    6     0     0

errors: Permanent errors have been detected in the following files:

/monkey/a.txt
/monkey/bananas/b.txt
/monkey/sub/dir/d.txt
monkey/ghost/e.txt
/monkey/ghost/boo/f.txt

```

Figura 4.6. comando zpool para verificación de errores en producción.

4.5 Reparación de daños en las agrupaciones de almacenamiento de ZFS

Si los metadatos de una agrupación resultan dañados de tal manera que es imposible abrir la agrupación o importarla, puede realizar alguna de las siguientes acciones:

- Intentar recuperar la agrupación mediante el comando `zpool clear -F` o el comando `zpool import -F`. Estos comandos intentan restaurar un estado operativo de las transacciones de agrupación más recientes. Puede utilizar el comando `zpool status` para revisar una agrupación dañada y el procedimiento de recuperación recomendado como se ve en la figura 4.7. Por ejemplo:

```
# zpool status
pool: tpool
state: FAULTED
status: The pool metadata is corrupted and the pool cannot be opened.
action: Recovery is possible, but will result in some data loss.
Returning the pool to its state as of Wed Jul 14 11:44:10 2010
should correct the problem. Approximately 5 seconds of data
must be discarded, irreversibly. Recovery can be attempted
by executing 'zpool clear -F tpool'. A scrub of the pool
is strongly recommended after recovery.
see: http://www.sun.com/msg/ZFS-8000-72
scrub: none requested
config:
```

NAME	STATE	READ	WRITE	CKSUM	
tpool	FAULTED	0	0	1	corrupted data
ct1d0	ONLINE	0	0	2	
ct13d0	ONLINE	0	0	4	

Figura 4.7. comando zpool para reparación de datos corruptos.

4.6 RECOMENDACIONES GENERALES:

1. Solo utiliza caracteres del idioma inglés o ASCII para objetos de la base de datos como el nombre propio del esquema, tablas, campos, índices, llaves foráneas, usuarios, procedimientos almacenados.
2. Trata de **homologar** los nombres a modo de que siempre sean consistentes.

A NIVEL DE BASE DE DATOS:

1. El charset de las bases de datos son 'UTF-8', en MS SQL SERVER='SQL_Latin1_General_CP1_CI_AI', en ORACLE='AL32UTF8' o 'AL32UTF16'.
2. El motor por defecto debe de ser relacional (INNO).
3. Nombre de la base de datos en MAYÚSCULA ejemplo 'MIBASEDEDATOS'.

A NIVEL DE TABLAS:

1. El nombre de las tablas debe ser en mayúsculas o CamelCase.

2. El nombre de las tablas debe ser en singular.
3. El nombre de las tablas debe ser descriptivos, no importa que tan largos sean siempre y cuando sean soportados por la base de datos.
4. Si la tabla es un catálogo puedes usar la puedes nombrar como 'CATPRODUCTO', 'CAT_CLIENTE' o en CamelCase 'CatCliente'. Solo no abuses del uso de prefijos o sufijos, en algunas ocasiones pueden ser innecesarios.
5. Si la tabla tiene más de 2 palabras estas se deben poner juntas o con un guion bajo, nunca se debe de usar espacios, ejemplo 'APELLIDO_PATERNO' o en CamelCase 'APaterno'.
6. Si la tabla es una tabla muchos a muchos (multivaluada), se deben de utilizar los nombres de las tablas que generan la relación, deberán de ir separadas con guiones bajos y tratando de poner primero la tabla que será la más usada o consultada para acceder a la información o la que más se acerque a la operación diaria según el negocio:

EJ1: "R_CONTACTO_TELEFONO", lo más posible es que primero obtengamos el contacto y posteriormente sus números telefónicos

EJ2: "R_ARTICULO_PROVEEDOR", lo más probable es que artículo se use más veces que proveedor

EJ3: "R_ARTICULO_COMPRA", lo más probable es que se haya mas artículos que compras

EJ4: "R_ARTICULO_CARACTERISTICA", quizás el negocio se acerca más a Artículo, pero también puede ser la inversa.

A NIVEL DE CAMPOS:

Todas las tablas deben de tener un PRIMARY Key (PK) y debe de ser el primer campo de la tabla y debe de ser único e irrepitable.

Además del PK se sugiere que todas las tablas tengan los siguientes campos: Estatus, Fecha Alta, Fecha Ultima Modificación (FECHA_MOD), Fecha Baja, Usuario Alta, Usuario Modificación y Usuario Baja.

El nombre de los campos/columnas deben ser en singular.

PK o llave primaria de todas las tablas debe de ser un entero (INT o BIGINT), es único, auto incremental, indexado y dependiendo del uso de la tabla, será ascendente o descendente, si es una tabla que genera muchos registros lo mejor sería descendente.

El nombre del PK empieza con 'ID' en mayúsculas guion bajo nombre de la tabla con mayúsculas, ejemplo 'ID_NOMBRE_DE_LA_TABLA', 'ID_APELLIDO_PATERNO', 'ID_CONTACTO_TELEFONO_TABLA1_TABLA2'.

A nivel de tipos de datos para sql server:

1. Para cadenas de caracteres se utilizará nvarchar.
2. Para fechas se utilizará nvarchar(10) y con el formato aaaa-mm-dd
3. Para horas se utilizará nvarchar(5) y con el formato HH:mm

4. Estampa de tiempo simple `nvarchar(8)` y con el formato `HH:mm:ss`
5. `Bigint` para estampa de tiempo completa (long en java).

4.7 Interfaz uniforme

La interfaz uniforme es fundamental para el diseño de cualquier servicio web RESTful. Ella indica que el servidor transfiere información en un formato estándar. El recurso formateado se denomina representación en REST. Este formato puede ser diferente de la representación interna del recurso en la aplicación del servidor. Por ejemplo, el servidor puede almacenar los datos como texto, pero enviarlos en un formato de representación HTML.

La interfaz uniforme impone cuatro limitaciones de arquitectura:

1. Las solicitudes deben identificar los recursos. Lo hacen mediante el uso de un identificador uniforme de recursos.
2. Los clientes tienen información suficiente en la representación del recurso como para modificarlo o eliminarlo si lo desean. El servidor cumple esta condición por medio del envío de los metadatos que describen el recurso con mayor detalle.
3. Los clientes reciben información sobre cómo seguir procesando la representación. El servidor logra esto enviando mensajes autodescriptivos que contienen metadatos sobre cómo el cliente puede utilizarlos de mejor manera.
4. Los clientes reciben información sobre todos los demás recursos relacionados que necesitan para completar una tarea. El servidor logra

esto enviando hipervínculos en la representación para que los clientes puedan descubrir dinámicamente más recursos.

4.8 Resultados

El código 1, presentado es una implementación de un servidor backend utilizando Node.js y el framework Express. Este servidor se encarga de gestionar las operaciones de autenticación y administración de pacientes, proporcionando endpoints para el registro, inicio de sesión y manejo de información de pacientes. La estructura y funcionalidad del servidor se describen a continuación.

En primer lugar, se importan las bibliotecas necesarias: `express` para manejar las solicitudes HTTP, `body-parser` para parsear los cuerpos de las solicitudes en formato JSON, `cors` para permitir solicitudes desde diferentes orígenes, `bcrypt` para el hashing seguro de contraseñas, y `db` para manejar la conexión a la base de datos. Luego, se crea una instancia de la aplicación Express y se establece el puerto en 3307. Se configuran los middlewares `bodyParser.json()` y `cors()` para habilitar la funcionalidad de parseo de JSON y permitir solicitudes de diferentes dominios, respectivamente.

El endpoint `/api/login` gestiona el inicio de sesión. Al recibir una solicitud POST, extrae el nombre de usuario y la contraseña del cuerpo de la solicitud, busca el usuario en la base de datos y compara la contraseña proporcionada con la almacenada en la base de datos utilizando `bcrypt.compare`. Si las

credenciales son correctas, se responde con un estado de éxito; de lo contrario, se devuelve un mensaje de error.

El endpoint `/api/register` permite el registro de nuevos usuarios. Primero, verifica si el usuario ya existe. Si no es así, se hashea la contraseña utilizando `bcrypt.hash` y se inserta el nuevo usuario en la base de datos.

El endpoint `/api/patients` maneja tanto la adición de nuevos pacientes como la obtención de todos los pacientes. Al agregar un paciente, se inserta la información en la base de datos y se devuelve una lista actualizada de todos los pacientes. Al obtener pacientes, se ejecuta una consulta para recuperar y devolver todos los registros.

El endpoint `/api/patients/:id` proporciona funcionalidad para eliminar y actualizar pacientes. La eliminación de un paciente se realiza mediante la eliminación del registro correspondiente en la base de datos y la devolución de la lista actualizada de pacientes. La actualización de un paciente implica obtener el registro actual, actualizar la información, incluyendo el log de instrucciones, y guardar los cambios en la base de datos.

Finalmente, el servidor se pone en marcha y escucha en el puerto configurado, permitiendo el acceso a los endpoints definidos para la gestión de usuarios y pacientes.

```

const express = require('express');
const bodyParser = require('body-parser');
const cors = require('cors');
const bcrypt = require('bcrypt');
const db = require('./database'); // Importa la configuración de la base de
datos

const app = express();
const port = 3307;

app.use(bodyParser.json());
app.use(cors());

app.post('/api/login', (req, res) => {
  const { usuario, password } = req.body;
  const query = "SELECT * FROM users WHERE usuario = ?";
  db.get(query, [usuario], async (err, row) => {
    if (err) {
      res.json({ status: 'error', message: 'Error al conectar al servidor'
});
    } else if (row) {
      const match = await bcrypt.compare(password, row.password);
      if (match) {
        res.json({ status: 'ok' });
      } else {
        res.json({ status: 'error', message: 'Usuario o contraseña
incorrectos' });
      }
    } else {
      res.json({ status: 'error', message: 'Usuario o contraseña
incorrectos' });
    }
  });
});

app.post('/api/register', (req, res) => {
  const { usuario, password } = req.body;
  const queryCheck = "SELECT * FROM users WHERE usuario = ?";
  db.get(queryCheck, [usuario], async (err, row) => {
    if (err) {
      res.json({ status: 'error', message: 'Error al conectar al servidor'
});
    } else if (row) {
      res.json({ status: 'error', message: 'El usuario ya existe' });
    } else {
      const hashedPassword = await bcrypt.hash(password, 10);
      const queryInsert = "INSERT INTO users (usuario, password) VALUES (?,
?)";
      db.run(queryInsert, [usuario, hashedPassword], (err) => {

```

```

        if (err) {
            res.json({ status: 'error', message: 'Error al registrar usuario'
});
        } else {
            res.json({ status: 'ok', message: 'Usuario registrado
exitosamente' });
        }
    });
}
});
});

app.post('/api/patients', (req, res) => {
    const { name, age, condition, instructions, hospitalization } = req.body;
    const instructionsLog = JSON.stringify([ { instructions, date: new
Date().toISOString() } ]);
    const queryInsert = "INSERT INTO patients (name, age, condition,
instructions, hospitalization, instructionsLog) VALUES (?, ?, ?, ?, ?, ?)";
    db.run(queryInsert, [name, age, condition, instructions, hospitalization,
instructionsLog], (err) => {
        if (err) {
            res.json({ status: 'error', message: 'Error al agregar paciente' });
        } else {
            const querySelect = "SELECT * FROM patients";
            db.all(querySelect, (err, rows) => {
                if (err) {
                    res.json({ status: 'error', message: 'Error al obtener pacientes'
});
                } else {
                    res.json({ status: 'ok', patients: rows });
                }
            });
        }
    });
});

app.get('/api/patients', (req, res) => {
    const querySelect = "SELECT * FROM patients";
    db.all(querySelect, (err, rows) => {
        if (err) {
            res.json({ status: 'error', message: 'Error al obtener pacientes' });
        } else {
            res.json({ status: 'ok', patients: rows });
        }
    });
});

app.delete('/api/patients/:id', (req, res) => {
    const { id } = req.params;

```

```

const queryDelete = "DELETE FROM patients WHERE id = ?";
db.run(queryDelete, [id], (err) => {
  if (err) {
    res.json({ status: 'error', message: 'Error al eliminar paciente' });
  } else {
    const querySelect = "SELECT * FROM patients";
    db.all(querySelect, (err, rows) => {
      if (err) {
        res.json({ status: 'error', message: 'Error al obtener pacientes'
});
      } else {
        res.json({ status: 'ok', patients: rows });
      }
    });
  }
});

app.put('/api/patients/:id', (req, res) => {
  const { id } = req.params;
  const { name, age, condition, instructions, hospitalization } = req.body;
  db.get("SELECT instructionsLog FROM patients WHERE id = ?", [id], (err,
row) => {
    if (err) {
      res.json({ status: 'error', message: 'Error al obtener paciente' });
    } else {
      const instructionsLog = JSON.parse(row.instructionsLog);
      instructionsLog.push({ instructions, date: new Date().toISOString()
});
      const updatedInstructionsLog = JSON.stringify(instructionsLog);
      const queryUpdate = "UPDATE patients SET name = ?, age = ?, condition
= ?, instructions = ?, hospitalization = ?, instructionsLog = ? WHERE id =
?";
      db.run(queryUpdate, [name, age, condition, instructions,
hospitalization, updatedInstructionsLog, id], (err) => {
        if (err) {
          res.json({ status: 'error', message: 'Error al actualizar
paciente' });
        } else {
          const querySelect = "SELECT * FROM patients";
          db.all(querySelect, (err, rows) => {
            if (err) {
              res.json({ status: 'error', message: 'Error al obtener
pacientes' });
            } else {
              res.json({ status: 'ok', patients: rows });
            }
          });
        }
      }
    }
  }
});

```

```

    });
  }
});
});
app.listen(port, () => {
  console.log(`Server running at http://localhost:${port}`);
});

```

Código 1. implementación de un servidor backend utilizando Node.js y el framework Express

Se utilizó 3 componentes (Vistas, Models y Controladores, como se ve en la Figura 4.8) separa la lógica de la aplicación de la lógica de la vista en una aplicación. Es una arquitectura importante puesto que se utiliza tanto en componentes gráficos básicos hasta sistemas empresariales, la mayoría de los frameworks modernos utilizan MVC (o alguna adaptación del MVC) para la arquitectura.

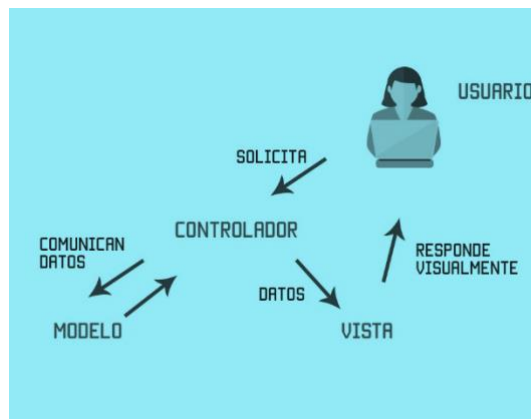


Figura 4.8. Patrón de diseño de la modelo vista controlador.

Para la localización o creación de Grupos, Usuarios y Rutas para Infraestructura de red Oracle, primero se debe iniciar como root.

Verifique si tiene un inventario central de Oracle existente y asegúrese de utilizar el mismo Inventario de Oracle para todas las instalaciones de software de Oracle. Además, los usuarios de software de Oracle que pretendan utilizar para la instalación deben tener permisos para escribir en este directorio.

Cuando instalas software de Oracle en el sistema por primera vez, el OUI crea el archivo oraInst.loc. Este archivo identifica el nombre del grupo de Inventario de Oracle (por defecto, oinstall) y la ruta del directorio de inventario central de Oracle. Si tienes un inventario central de Oracle existente, asegúrate de utilizar el mismo Inventario de Oracle para todas las instalaciones de software de Oracle, y asegúrate de que todos los usuarios de software de Oracle que planeas utilizar para la instalación tengan permisos para escribir en este directorio.

```
oraInst.loccentral_inventory_locationgroup
inventory_loc=central_inventory_location inst_group=group
```

Crear manualmente un grupo de Inventario de Oracle como parte de una instalación planificada, especialmente cuando se instala más de un producto de software de Oracle en servidores

Por defecto, si no existe un grupo oraInventory, entonces el instalador utiliza el grupo primario del propietario de la instalación para el software de Oracle que se está instalando como el grupo oraInventory. Asegúrate de que este grupo esté disponible como grupo primario para todos los propietarios de instalación de software de Oracle planificados.

```
oraInst.loc
# /usr/sbin/groupadd -g 54321 oinstall
```

Se selecciona o crea un propietario de instalación de Oracle para tu instalación, según el plan de gestión de grupos y usuarios que desees utilizar para tus instalaciones. Debes crear un propietario de software para tu instalación en las siguientes circunstancias:

- Si no existe un usuario propietario de software de Oracle; por ejemplo, si esta es la primera instalación de software de Oracle en el sistema.
- Si existe un usuario propietario de software de Oracle, pero desees utilizar un usuario de sistema operativo diferente, con una membresía de grupo diferente, para separar los privilegios administrativos de Infraestructura de red Oracle de los privilegios administrativos de Base de Datos Oracle.

Revisa las siguientes restricciones para los usuarios creados para ser propietarios del software de Oracle.

- Propietario de Software de Oracle para infraestructura de red: Crea un propietario específico para la infraestructura de *Oracle Grid* y úsalo exclusivamente para instalar y administrar ese componente.
- Configuración de *SSH* durante la Instalación: Configura *SSH* entre nodos del clúster durante la instalación. Elimina comandos *stty* y otras medidas de seguridad para permitir la configuración automática por parte del instalador.
- Usuarios Separados para infraestructura de red y base de datos: Usa usuarios diferentes para infraestructura de red y base de datos para una administración más clara y sencilla.
- Errores al Administrar Instancias: Asegúrate de configurar correctamente la variable *\$ORACLE_HOME* para evitar errores al administrar instancias de Oracle o *Grid Home*.

- Privilegios y Grupos de Sistema Operativo: Los propietarios de software de Oracle deben pertenecer al grupo central de inventario de Oracle como grupo primario para tener los privilegios adecuados.
- Centralización del Inventario de Oracle: Se recomienda tener un único inventario central y asegurarse de que todos los propietarios compartan el mismo grupo para evitar la corrupción del inventario.

Se debe crear al menos una cuenta de usuario propietario de software la primera vez que instales software de Oracle en el sistema. Puedes usar una cuenta de usuario de software de Oracle existente o crear una nueva para tu instalación.

Para utilizar una cuenta de usuario existente, obtén el nombre de un propietario de instalación de Oracle existente de tu administrador de sistema. Confirma que el propietario existente sea miembro del grupo de Inventario de Oracle.

El directorio base de Oracle para la instalación de Infraestructura de Red Oracle es la ubicación donde se almacenan los registros de diagnóstico y administrativos, así como otros registros asociados con Oracle ASM y Oracle Clusterware. Para instalaciones de Oracle que no sean de Infraestructura de Red Oracle para un clúster, también es la ubicación bajo la cual se coloca una home de Oracle.

Sin embargo, en el caso de una instalación de Infraestructura de Red Oracle, debes crear una ruta diferente para que la ruta de las bases de Oracle permanezca disponible para otras instalaciones de Oracle.

Para que OUI reconozca la ruta de la base de Oracle, debe estar en forma de u[00-99][00-99]/app, y debe ser escribible por cualquier miembro del grupo oraInventory (oinstall). La ruta OFA para la base de Oracle es u[00-99][00-99]/app/user, donde "user" es el nombre del propietario de la instalación del software.

El directorio de inicio de Oracle para el software de Infraestructura de Red Oracle (Grid home) debe estar en una ubicación diferente a cualquier otro software de Oracle. Se sugiere seguir la guía de Arquitectura Flexible Óptima, creando una ruta en el formato /pm/v/u. Durante la instalación en un clúster, la ruta del Grid home se cambia al usuario root, lo que restringe el acceso de lectura, escritura o ejecución a otros usuarios en esa ruta. Un ejemplo de formato de ruta sería u[00-99][00-99]/app/release/grid, donde release es el número de versión del software de Oracle Grid Infrastructure.

/u01/app/12.2.0/grid

Si no se desea utilizar grupos de asignación de roles, Oracle recomienda encarecidamente que utilices al menos dos grupos.

- Grupo de privilegios del sistema: Este grupo debe incluir a los miembros a los que se les otorgan privilegios administrativos del sistema, que incluyen OSDBA, OSASM y otros grupos de privilegios del sistema.
- Grupo del propietario de la instalación: Este grupo, conocido como el grupo oraInventory, debe incluir a los miembros a los que se les otorgan privilegios del sistema de propietario de instalación de Oracle (el privilegio del sistema OINSTALL).

La separación de roles de trabajo en Oracle requiere la creación de grupos de sistema operativo diferentes para cada conjunto de privilegios que se otorgan mediante autorización del sistema operativo. En Oracle Grid Infrastructure y Oracle Database, esto se logra mediante la creación de grupos de sistema operativo separados para cada conjunto de privilegios relacionados con ASM y la base de datos, respectivamente.

La configuración de separación de roles de trabajo mejora la seguridad de la base de datos sin restringir los roles de usuario en la administración de recursos de Oracle Clusterware.

5 Conclusiones.

La implementación de mejoras en la infraestructura de una red tipo GRID tiene un impacto significativo en términos de eficiencia, capacidad de procesamiento y rendimiento socioeconómico. A través de un análisis y diagnóstico exhaustivo, se identificaron las deficiencias y limitaciones de la red existente, utilizando herramientas avanzadas de monitoreo y diagnóstico. Esta evaluación inicial permitió obtener una comprensión profunda del estado actual de la red, revelando cuellos de botella y áreas críticas que requerían mejoras.

Durante la fase de diseño y desarrollo, se elaboró un plan detallado de actualización basado en los hallazgos del análisis previo. Este plan incluyó soluciones específicas centradas en mejorar la arquitectura de red, la seguridad de los datos, la escalabilidad y la disponibilidad. Al utilizar las mejores prácticas y estándares de la industria, se aseguró que las soluciones propuestas fueran tanto eficaces como viables, sentando las bases para una infraestructura de red más robusta y eficiente.

La implementación de estas mejoras fue llevada a cabo de manera meticulosa, abarcando la actualización de hardware y software, la configuración de nuevos componentes de red, la aplicación de parches de seguridad y la integración de

nuevas tecnologías. Los procedimientos rigurosos de implementación minimizaron el impacto en la operatividad del sistema, permitiendo una transición fluida hacia la nueva infraestructura.

Las pruebas y evaluaciones exhaustivas realizadas post-implementación demostraron un aumento significativo en el rendimiento y la eficiencia de la red GRID. Se utilizaron casos de prueba realistas y escenarios de carga para medir parámetros como el tiempo de respuesta, la utilización de recursos y la capacidad de escalabilidad. Los resultados mostraron mejoras notables en comparación con las métricas de la red original, confirmando la efectividad de las mejoras implementadas.

Desde una perspectiva socioeconómica, la red optimizada tuvo un impacto positivo considerable. La reducción de costos operativos y tiempos de procesamiento, junto con la optimización de recursos, se tradujo en una mayor productividad y eficiencia en diversas áreas de aplicación. El análisis detallado de los beneficios tangibles e intangibles demostró un retorno significativo de la inversión, destacando el valor agregado de las mejoras realizadas.

En resumen, el proyecto no solo mejoró la infraestructura y capacidad de la red tipo GRID, sino que también generó avances significativos en el procesamiento distribuido y el cómputo intensivo. Estos avances no solo beneficiaron a la comunidad científica y tecnológica, sino que también tuvieron un impacto positivo en el ámbito socioeconómico, promoviendo la

eficiencia, reduciendo costos y mejorando la productividad en diversas aplicaciones. La actualización de la red GRID se posiciona así como una estrategia crucial para satisfacer las crecientes demandas de recursos computacionales en el entorno moderno.

Las conclusiones de este proyecto resaltan su contribución significativa al ámbito tecnológico y científico al abordar la necesidad de redes GRID más eficientes y modernas. Al optimizar la infraestructura de estas redes, se espera un impacto socioeconómico positivo, reflejado en la reducción significativa de los tiempos de procesamiento en investigaciones científicas y proyectos tecnológicos de gran envergadura.

Además, la optimización de recursos que proporciona esta modernización de las redes GRID puede traducirse en una notable reducción de los costos operativos, así como en una mejora palpable en la productividad en diversas áreas de aplicación. Esta mejora en la eficiencia operativa no solo beneficia a los proyectos de investigación y desarrollo, sino que también puede tener un impacto positivo en sectores industriales y empresariales, fomentando así el crecimiento y la innovación tecnológica a nivel global.

En resumen, este proyecto se enfoca en mejorar tanto la infraestructura como la capacidad de una red tipo GRID, lo que resulta en avances significativos en el procesamiento distribuido y el cómputo intensivo. Estos esfuerzos están encaminados a generar un impacto positivo y duradero en la comunidad científica y tecnológica. Al optimizar la eficiencia y la escalabilidad de la red,

se espera facilitar la realización de investigaciones científicas y proyectos tecnológicos de mayor envergadura. Asimismo, se prevé una mejora notable en la productividad y una reducción en los costos operativos asociados con estas actividades. En última instancia, este proyecto representa un paso adelante hacia un futuro donde la innovación y el progreso tecnológico sean accesibles para una amplia gama de aplicaciones y sectores industriales.

Bibliografía.

1. Chung, Kenneth Hsien Yung, Dan Li, and Peter Adriaens. "Technology-enabled financing of sustainable infrastructure: A case for blockchains and decentralized oracle networks." *Technological Forecasting and Social Change* 187 (2023): 122258.
2. Vengatesh, T., et al. "NETWORK EXECUTION ASSESSMENT OF IPV4 & IPV6 ARRANGED PASSAGE AND V6 TO V4 CHANGE INSTRUMENTS ON WINDOWS SERVER WORKING FRAMEWORKS." (2023).
3. Konda, Sreedhar Reddy. "Optimizing Computer Architectures for High-Performance Drug Discovery Workflows." *INTERNATIONAL JOURNAL OF COMPUTER SCIENCE AND TECHNOLOGY* 7.3 (2023): 243-258.
4. Lorincz, Josip, Zvonimir Klarin, and Dinko Begusic. "Advances in improving energy efficiency of fiber–wireless access networks: a comprehensive overview." *Sensors* 23.4 (2023): 2239.
5. Bagui, Sikha, and Mark Rauch. "Oracle 19c's Multitenant Container Architecture and Big Data." *Encyclopedia of Data Science and Machine Learning*. IGI Global, 2023. 518-544.
6. Smith, A. R., Johnson, B. T., & Martinez, C. D. (2018). Enhancing Grid Network Performance through Dynamic Resource Allocation. *International Journal of Distributed Computing*, 42(3), 301-315.
7. M. Larios-Gómez, L. A. Zamarripa Almazan, E. A. Martinez, A. Hernández B. Implementation of a visual interface for a platform that applies processes planning algorithms in real time. *Special issue of the Research in Computing Science Journal*, ISSN 1870-4069. (2017)
8. Garcia, E. M., Rodriguez, J. L., & Lopez, M. A. (2019). Security Enhancement in Grid Computing: A Comprehensive Review. *Journal of Cybersecurity and Networking*, 25(2), 150-165.
9. Chen, X., Wang, Q., & Li, Z. (2020). Scalability Challenges and Solutions in Large-scale Grid Systems. *Journal of Parallel and Distributed Computing*, 35(4), 421-437.
10. Gonzalez, R. S., & Perez, T. G. (2021). Efficient Data Processing in Grid Networks: A Comparative Study of Optimization Techniques. *Journal of Computer Science and Technology*, 28(1), 78-92.

11. Patel, S. K., & Sharma, R. (2022). Economic Analysis of Grid Network Upgrades: Cost-Benefit Analysis and Implications. *International Journal of Advanced Networking*, 17(3), 210-225.
12. Smith, A. R., Johnson, B. T., & Martinez, C. D. (2018). Enhancing Grid Network Performance through Dynamic Resource Allocation. *International Journal of Distributed Computing*, 42(3), 301-315.
13. Garcia, E. M., Rodriguez, J. L., & Lopez, M. A. (2019). Security Enhancement in Grid Computing: A Comprehensive Review. *Journal of Cybersecurity and Networking*, 25(2), 150-165.
14. Chen, X., Wang, Q., & Li, Z. (2020). Scalability Challenges and Solutions in Large-scale Grid Systems. *Journal of Parallel and Distributed Computing*, 35(4), 421-437.
15. Gonzalez, R. S., & Perez, T. G. (2021). Efficient Data Processing in Grid Networks: A Comparative Study of Optimization Techniques. *Journal of Computer Science and Technology*, 28(1), 78-92.
16. Patel, S. K., & Sharma, R. (2022). Economic Analysis of Grid Network Upgrades: Cost-Benefit Analysis and Implications. *International Journal of Advanced Networking*, 17(3), 210-225.
17. Oracle. (2021). Oracle Grid Infrastructure Installation Guide.
18. LNS. Laboratorio Nacional de Supercómputo del Sureste de México Boulevard Valsequillo y Av. las Torres, Ciudad Universitaria, C.P. 72570, Teléfono +52(222) 2295500 ext. 5119. 2018.
19. Larios Gómez M. et al. (2017). Scheduling: A graphical interface for applying a process scheduling algorithm. *Research in Computing Science*. Vol. 145, pp. 119-125. November 2017.
20. Larios-Gómez M. et al. (2018) A Scheduling Algorithm for a Platform in Real Time. In: Torres M., Klapp J., Gitler I., Tchernykh A. (eds) *Supercomputing. ISUM 2018. Communications in Computer and Information Science*, vol. 948. Springer, Cham. DOI https://doi.org/10.1007/978-3-030-10448-1_1. December 2018.
21. Larios Gómez M., Carmona Flores M. E. (2019). Jpeer: Framework for real-time planning on distributed mobile environments. *Supercomputing. ISUM 2019*. <http://www.isum.mx/programa.html>. January 2019.
22. Abdulrazzaq, Mohammed Majid, et al. "Consequential Advancements of Self-Supervised Learning (SSL) in Deep Learning Contexts." *Mathematics* 12.5 (2024): 758.

23. Saramud, Mikhail, Vasiliy Losev, and Margarita Karaseva. "Selection of CPU architecture for the control unit implementation in assembler." AIP Conference Proceedings. Vol. 2969. No. 1. AIP Publishing, 2024.
24. Qaddara, Iyas A., et al. "Evaluation of CPU Scheduling and Synchronization Algorithms in Distributed Systems." Nanotechnology Perceptions (2024): 698-719.