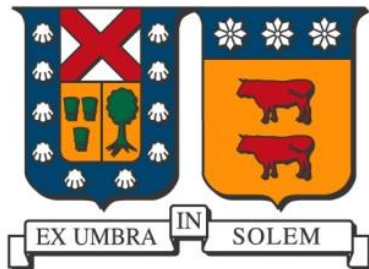


UNIVERSIDAD TÉCNICA FEDERICO SANTA MARÍA
VIÑA DEL MAR – CHILE



**“SISTEMA CENTRALIZADO DE AUTOMATIZACIÓN
PARA LA GESTIÓN SEGURA DE EQUIPAMIENTO
DE RED y SERVIDORES”**

MAXIMILIANO VICENTE SOTO SEPÚLVEDA

TRABAJO DE TITULACIÓN PARA OPTAR AL TÍTULO DE INGENIERO EN
INFORMÁTICA/INGENIERÍA EN INFORMÁTICA

PROFESOR GUÍA: GONZALO MENDOZA CÁRDENAS

PROFESOR CORREFERENTE: RODRIGO PINOCHET

ABRIL – 2026



CONSTANCIA DE VALIDACIÓN Y CONFIDENCIALIDAD DE MONOGRAFÍA A REPOSITORIO ACADÉMICO

1.- IDENTIFICACIÓN DEL TRABAJO ACADÉMICO

Tipo de monografía (marcar una opción): ☒ Memoria o trabajo de título ☐ Tesis de Postgrado

Título del trabajo: Sistema centralizado de automatización para la gestión segura de equipamiento de red y servidores

Nombre del candidato(a): Maximiliano Soto Sepúlveda

Carrera / Grado: Ingeniería en Informática

Campus: Sede Viña del Mar **Departamento:** _____

2.- VALIDACIÓN DEL PROFESOR GUÍA/DIRECTOR DE TESIS

Yo, Gonzalo Mendoza Cárdenas, en mi calidad de profesor(a) guía/director(a) del trabajo académico mencionado anteriormente **DEJO CONSTANCIA** que:

- He revisado esta versión del documento y corresponde a la versión final aprobada del trabajo.
- El trabajo cumple con los requisitos académicos y de formato establecidos por la institución.

3.- EVALUACIÓN DE CONFIDENCIALIDAD POR PROPIEDAD INDUSTRIAL (marcar una opción)

☒ El trabajo **NO contiene** información que amerite confidencialidad y puede ser publicado de inmediato en repositorio con acceso abierto.

☐ El trabajo **CONTIENE** información con potenciales implicancias de propiedad industrial o intelectual y requiere un periodo de confidencialidad (**embargo**) por (**marcar una opción**):

☐ 6 meses ☐ 12 meses ☐ 2 años ☐ 3 años ☐ 5 años ☐ 10 años

Fundamentación de la necesidad de confidencialidad (obligatorio si se solicita embargo):

4.- FIRMAS

Profesor(a) guía o director(a) de memoria o tesis:

Fecha: 07/09/2026 **Firma:** _____

Estudiante o Candidato(a):

Fecha: 17/08/2026 **Firma:** Maximiliano Soto Sepúlveda

Este formulario debe ser insertado como página 2 de la memoria o tesis, completado y firmado por estudiante y profesor(a) antes de la entrega en portal PRISMA de Biblioteca USM.

DEDICATORIA

Este trabajo está dedicado con profundo amor y gratitud, principalmente a mis padres, Raúl Soto y Lily Sepúlveda, quienes han sido el pilar fundamental a lo largo de todo mi proceso académico y personal. Su apoyo incondicional y sus sacrificios han hecho posible la culminación de este proyecto.

De igual manera, extendiendo esta dedicatoria a mis amistades y seres queridos que me han acompañado y brindado su aliento en este camino.

RESUMEN

Resumen—El presente trabajo implementa un sistema centralizado de automatización utilizando Ansible para la gestión segura de una infraestructura híbrida de servidores y equipos de red. El proyecto aborda la problemática de la fricción operacional, entendida como la ineficiencia y lentitud causada por procesos manuales repetitivos que elevan el riesgo de error humano, transformando la administración tradicional en un modelo de Infraestructura como Código (IaC).

La metodología se centró en la implementación iterativa de la Base Funcional (L2/L3) y el diseño de Playbooks dedicados al *hardening* y la estandarización.

Los resultados demuestran una mejora cuantitativa superior frente al modelo tradicional:

- Eficiencia (TDC y TEC): Se certificó una reducción superior al 90% en el Tiempo de Despliegue de Configuración (TDC), con la validación de una Tasa de Error de Configuración (TEC) de cercana al 0%.
- Resiliencia (TRS): El monitoreo proactivo de los servicios críticos (DNS/DB) redujo el Tiempo Medio de Detección (MTTD) y confirmó el cumplimiento del umbral de Tiempo de Recuperación de Servicio (TRS) < 10 segundos.

El proyecto valida que la automatización no solo optimiza el tiempo, sino que elimina el riesgo humano y eleva el estándar de seguridad de la infraestructura a niveles de calidad inalcanzables con métodos tradicionales.

Palabras Clave—Automatización; Ansible; *Hardening*; Trazabilidad; Infraestructura como Código; Gestión de Red.

GLOSARIO

Agentes: Modelo de automatización que requiere la instalación de un software persistente en cada nodo gestionado.

Agentless: Modelo de automatización donde no se requiere la instalación de software permanente en los nodos gestionados.

Ansible: Plataforma de automatización de TI de código abierto que permite automatizar el aprovisionamiento, la configuración, la implementación de software y la orquestación de flujos de trabajo complejos en entornos de TI.

Automatización: Uso de sistemas y software para llevar a cabo tareas repetitivas y rutinarias sin necesidad de intervención humana continua. Permitiendo aumentar la productividad, reducir errores y optimizar procesos.

Backups: Copias de los datos originales de un sistema, realizadas y almacenadas en un lugar seguro, con el objetivo de poder recuperar esa información en caso de que los datos originales se pierdan, corrompan o sean inaccesibles.

Backbone: Parte central y principal de una red que maneja el tráfico más pesado y proporciona la ruta principal de conectividad entre las distintas subredes, segmentos (VLANs), o áreas geográficas.

CLI: Mecanismo de interacción con un sistema operativo o un programa de software mediante la escritura de líneas de texto (comandos)

Hardening: Conjunto de prácticas de ciberseguridad para fortalecer un sistema informático, red o dispositivo, reduciendo su superficie de ataque mediante la eliminación de servicios y configuraciones innecesarias, el cierre de puertos no utilizados y la aplicación de parches y permisos de seguridad adecuados.

Interfaces Virtuales de Switch (SVIs): interfaces lógicas (no físicas) que se crean en un *switch* de Capa 3 para funcionar como el *Gateway* (puerta de enlace) predeterminado para una VLAN específica.

L2 (Capa de Enlace de Datos): Nivel del modelo OSI encargado de la transmisión de datos entre dispositivos dentro de una misma red local. Gestiona el direccionamiento físico mediante direcciones MAC, el control de acceso al medio y la detección de errores.

L3 (Capa de Red): Nivel del modelo OSI responsable del direccionamiento lógico y el enrutamiento de los datos entre diferentes redes. Utiliza direcciones IP para determinar la mejor ruta que deben seguir los paquetes hasta su destino.

L4 (Capa de Transporte): Nivel del modelo OSI encargado de la comunicación extremo a extremo, utilizando protocolos como TCP y UDP.

L7 (Capa de Aplicación): Nivel del modelo OSI que interactúa directamente con las aplicaciones, permitiendo inspección y control del tráfico basado en servicios y contenido.

LAN: Red que conecta dispositivos que se encuentran en un área geográfica limitada y pequeña.

Logs: Archivos generados automáticamente por un sistema operativo, una aplicación, un servidor o un dispositivo de red que documentan eventos y actividades que ocurren dentro de ese sistema.

Loops: Bucle que ocurre cuando hay múltiples caminos físicos o redundantes entre dos dispositivos de red sin un protocolo activo que gestione o bloquee los caminos duplicados

Nodo: Dispositivo o máquina que se puede administrar o al que se puede desplegar código de automatización.

On-premise: Sistema informático donde los recursos, como servidores y almacenamiento, están ubicados físicamente en las instalaciones de la organización.

Ping: Utilidad de software de red que se utiliza para probar la accesibilidad de un host (servidor o dispositivo) en una red IP y para medir el tiempo de ida y vuelta que tardan los paquetes en ser recibidos de nuevo.

Playbooks: Archivos escritos en formato YAML que funcionan como "listas de tareas" o "guiones" para automatizar la configuración, el despliegue y la orquestación de sistemas en entornos de TI

PNETlab: Plataforma de simulación de redes que permite emular dispositivos como routers, switches y firewalls para crear y probar topologías de red.

Porcentaje de Convergencia de Protocolo: Mide el porcentaje de tiempo que los protocolos de redundancia permanecen en un estado estable y sincronizado (Master / in-sync). La meta es mantener el 100% para asegurar la estabilidad del *backbone* de la red.

Ransomware: Es un tipo de programa malicioso que bloquea o cifra los archivos de un usuario o sistema, impidiendo su acceso, y exige el pago de un rescate a cambio de la clave para descifrarlos o eliminar la restricción.

Rollback: Proceso de deshacer o revertir los cambios realizados en un sistema, software, base de datos o configuración para restaurarlos a un estado funcional y estable anterior.

Routing: Proceso de seleccionar la mejor ruta o camino para que los paquetes de datos viajen a través de una o varias redes, desde el origen hasta el destino. Este proceso opera en la Capa 3 del Modelo OSI

Switching: Proceso de filtrar y reenviar tramas de datos entre puertos dentro de una misma red local (LAN). Opera en la Capa 2 del Modelo OSI

Tasa de Error de Configuración (TEC): Mide el porcentaje de fallos o errores de sintaxis/lógica que ocurren durante la aplicación de la configuración. El objetivo del modelo de Infraestructura como Código (IaC) es alcanzar el 0% de error, eliminando la probabilidad de inestabilidad humana y garantizando la calidad.

Tasa de Éxito de Respaldos de Configuración: Mide el porcentaje de veces que el proceso automatizado de *backup* de configuraciones se completa sin fallos. El objetivo es alcanzar el 100%, asegurando la fiabilidad y la reducción del riesgo de pérdida de configuración.

Tiempo de Despliegue de Configuración (TDC): Métrica que mide el tiempo total requerido para aplicar una configuración estandarizada a la infraestructura, desde el inicio de la ejecución hasta la finalización de la tarea. La automatización busca una reducción superior al 90% del tiempo en comparación con el método manual.

Tiempo Medio de Recuperación (MTTR): Métrica de fiabilidad que representa el tiempo promedio que tarda el sistema o un componente en ser restaurado a su estado operativo normal después de una falla. En el contexto de TRS, MTTR se utiliza a menudo como el término estadístico o promedio que engloba el tiempo total de diagnóstico y reparación (TRS).

Tiempo de Recuperación de Servicio (TRS): Mide el tiempo total que transcurre desde que se identifica una falla crítica en el sistema hasta que el servicio afectado vuelve a estar completamente operativo y funcional.

Topología: Diseño o disposición de los elementos de una red informática, que puede ser tanto física como lógica. Determina la estructura de la red, su rendimiento, eficiencia y resiliencia.

VLAN: Red lógica separada dentro de una red física. Los dispositivos que están en la misma VLAN pueden comunicarse entre sí como si estuvieran conectados a una red física separada.

VMware Workstation: Software de virtualización que permite crear y gestionar múltiples máquinas virtuales en un computador físico, facilitando la ejecución de diferentes sistemas operativos simultáneamente.

INDICE DE CONTENIDOS

1. INDICE DE FIGURAS.....	9
2. INDICE DE TABLAS.....	10
3. INTRODUCCIÓN	11
CAPÍTULO 1: DEFINICIÓN DEL PROBLEMA	12
1.1.DESCRIPCIÓN DEL PROBLEMA	12
1.2. OBJETIVO GENERAL.....	13
1.3. OBJETIVOS ESPECÍFICOS	13
4. CAPÍTULO 2: MARCO TEÓRICO	14
2.1. CONCEPTOS GENERALES DE INFRAESTRUCTURA DE RED ...	14
2.2. AUTOMATIZACIÓN EN INFRAESTRUCTURA.....	14
2.3. SEGURIDAD EN LA INFRAESTRUCTURA	14
2.4. MONITOREO Y RESPALDO DE CONFIGURACIONES	15
2.5. ANÁLISIS CUALITATIVO Y CUANTITATIVO DE ALTERNATIVAS DE SOLUCIÓN.....	15
2.6. SINTESIS DE LA SOLUCIÓN Y CRITERIOS DE IMPLEMENTACIÓN	18
5. CAPÍTULO 3: PROPUESTA DE SOLUCIÓN.....	19
3.1. PLAN DE TRABAJO	21
ETAPA 1: ANÁLISIS Y DISEÑO LÓGICO	23
ETAPA 2: IMPLEMENTACIÓN BASE (L2/L3)	23
ETAPA 3: IMPLEMENTACIÓN DEL SISTEMA DE AUTOMATIZACIÓN	25
ETAPA 4: APLICACIÓN DE MEDIDAS DE SEGURIDAD	27
ETAPA 5: MONITOREO, RESPALDO Y VALIDACIÓN	30
ETAPA 6: DOCUMENTACIÓN Y CONCLUSIONES.....	32
6. CAPÍTULO 4: VALIDACIÓN DE LA SOLUCIÓN	34
4.1. ESCENARIO DE LA COMPLEJIDAD DE LA GESTIÓN MANUAL	34
4.2. EL VALOR DE LA GESTIÓN AUTOMATIZADA	35
7. CAPÍTULO 5: CONCLUSIONES.....	40
8. REFERENCIAS BIBLIOGRÁFICAS	42

INDICE DE FIGURAS

Figura 1. Diseño de Topología Propuesto	20
Figura 2. Configuración de Gateways (SVIs) y Segmentación Lógica en ambos Switch Core (Redundancia).....	24
Figura 3. Validación de Servicio ssh Activo.....	25
Figura 4. Flujo de Ejecución de Playbooks.....	26
Figura 5. Ejemplo de Estructura de Playbook YAML (Tarea de Hardening)	28
Figura 6. Evidencia Cuantitativa de Hardening (TDC y TEC)	35
Figura 7. Ejecución de Playbook de Verificación Proactiva de Servicios Críticos (Reducción del MTTD).....	36
Figura 8. Evidencia Cuantitativa de Hardening: Reporte de Auditoría y Verificación de Cumplimiento (TEC 0%).....	37
Figura 9. Evidencia de Verificación Proactiva de Servicios Críticos (DNS/DB).....	38

INDICE DE TABLAS

Tabla 1 Comparativa Herramientas de Automatización	16
Tabla 2 Evaluación de la Optimización Operacional (KPIs).....	17
Tabla 3 Plan de trabajo	22
Tabla 4 Análisis Cuantitativo y Umbrales de Funcionamiento	31
Tabla 5 Comparativa de Resultados: Gestión Manual vs. Automatización.	39

INTRODUCCIÓN

En el escenario informático actual, dos aspectos resultan esenciales para la gestión eficiente de la infraestructura tecnológica: la optimización operativa y la seguridad de la información.

Por una parte, la optimización busca que las tareas rutinarias de administración y gestión de infraestructura puedan ser automatizadas, permitiendo que se destine más tiempo a otras labores como la investigación, la innovación y el desarrollo de nuevas tecnologías. De esta manera, la automatización se consolida como una herramienta clave para mejorar la eficiencia operativa, disminuir la carga de trabajo repetitiva y minimizar los errores derivados de la intervención humana.

Por otra parte, la seguridad se ha convertido en una prioridad ante el crecimiento sostenido de incidentes como ciberataques, infecciones por *ransomware*, vulnerabilidades no parchadas y políticas de control de acceso mal implementadas. Una infraestructura con configuraciones inconsistentes, débiles o no estandarizadas puede convertirse en un punto de exposición crítica, comprometiendo la disponibilidad, la integridad y la confidencialidad de los sistemas de red y servidores.

En este contexto, surge la necesidad de implementar una solución que permita mitigar estas vulnerabilidades y, al mismo tiempo, optimizar la administración de los recursos informáticos.

El presente informe presenta el diseño e implementación de un sistema centralizado de automatización, orientado a optimizar la administración de la infraestructura tecnológica y fortalecer los mecanismos de seguridad, contribuyendo así a una gestión más confiable, consistente y resiliente de los recursos informáticos.

CAPÍTULO 1: DEFINICIÓN DEL PROBLEMA

1.1. DESCRIPCIÓN DEL PROBLEMA

La gestión y administración de los equipos de red y servidores son fundamentales para garantizar el correcto funcionamiento, la estabilidad y disponibilidad de cualquier infraestructura tecnológica. Sin embargo, en numerosos entornos organizacionales estos procesos continúan realizándose de manera manual, lo que genera procedimientos ineficientes, lentos y altamente susceptibles a errores humanos. Esta dependencia de la intervención manual no solo incrementa el margen de fallas operativas, sino que también dificulta la estandarización y el control de las configuraciones.

A este escenario se le suma el desafío de la seguridad informática, donde configuraciones débiles, servicios innecesarios habilitados o políticas de acceso mal definidas representan vulnerabilidades que pueden comprometer la confidencialidad, integridad y disponibilidad de la información. Dichas debilidades incrementan la exposición frente a ciberataques, infecciones por *ransomware* o accesos no autorizados, que pueden derivar en pérdida de datos o indisponibilidad de servicios críticos.

Paralelamente, el monitoreo de la infraestructura se convierte en un aspecto clave para la continuidad operativa. Una infraestructura carente de mecanismos de supervisión y control proactivo tiende a experimentar mayores tiempos de indisponibilidad y menor capacidad de respuesta ante incidentes. Los servicios tecnológicos suelen medirse en porcentajes de disponibilidad, donde alcanzar niveles cercanos al 99% depende directamente de la existencia de sistemas de monitoreo eficientes que permitan la detección temprana de fallas y la aplicación oportuna de acciones correctivas.

En contraste, una infraestructura que dispone de un monitoreo adecuado y mecanismos automatizados de gestión es capaz de identificar, analizar y mitigar incidentes de manera más efectiva, reduciendo los tiempos de recuperación y manteniendo la continuidad operativa de los servicios ofrecidos a los usuarios.

1.2. OBJETIVO GENERAL

Diseñar e implementar una solución centralizada de automatización que permita la configuración y gestión de la infraestructura de red de manera eficiente, incorporando mecanismos de seguridad, optimización de tiempos de despliegue y estandarización en los procesos de administración.

1.3. OBJETIVOS ESPECÍFICOS

1. Diseñar un modelo de red organizado y estandarizado, identificando los servicios críticos y estableciendo los requisitos de continuidad operacional y seguridad que minimicen los procesos ineficientes y las configuraciones propensas a error.
2. Estandarizar configuraciones de equipos de red y servidores, garantizando coherencia, trazabilidad y reducción de errores en la aplicación de cambios.
3. Diseñar políticas de seguridad básicas, orientadas a minimizar opciones de ataque mediante la inhabilitación de puertos y servicios innecesarios, controlando el acceso y asegurando la integridad de la infraestructura.
4. Establecer un sistema de monitoreo y verificación automatizado, que permita detectar variaciones o fallos de manera preventiva, contribuyendo a la disponibilidad y estabilidad de los servicios.
5. Evaluar el impacto de la solución propuesta, mediante la comparación de métricas operativas antes y después de la automatización, midiendo mejoras en tiempos de implementación, reducción de errores y fortalecimiento de la seguridad.

CAPÍTULO 2: MARCO TEÓRICO

2.1. CONCEPTOS GENERALES DE INFRAESTRUCTURA DE RED

La infraestructura de red constituye el conjunto de dispositivos, servicios y tecnologías que permiten la comunicación, el intercambio de datos y la continuidad de los servicios dentro de una organización. Está conformada principalmente por equipos de red como *routers*, *switches*, *firewalls* además de servidores encargados de gestionar recursos y servicios críticos [1]. No solo consta de dispositivos que establecen la comunicación, sino que estos elementos son esenciales para gestionar, asegurar y administrar el flujo de datos. Su función principal es mantener la red operativa, asegurando el intercambio eficiente de información.

2.2. AUTOMATIZACIÓN EN INFRAESTRUCTURA

La automatización consiste en ejecutar tareas de configuración o administración de forma automática, mediante herramientas o *scripts* que reemplazan la intervención manual. Este enfoque, conocido como Infraestructura como Código (*IaC*) [2], permite aplicar configuraciones uniformes, disminuir errores y mejorar la trazabilidad de los cambios.

La automatización es fundamental para gestionar, cambiar y adaptar tu infraestructura de TI y la forma en que funciona tu empresa. Los equipos de TI pueden automatizar procesos complejos para aumentar la eficiencia, productividad y flexibilidad, al mismo tiempo que reducen costes y errores humanos. [3]

2.3. SEGURIDAD EN LA INFRAESTRUCTURA

La seguridad informática busca proteger los recursos tecnológicos garantizando confidencialidad, integridad y disponibilidad [4]. En redes y servidores, las configuraciones débiles o los servicios innecesarios representan vulnerabilidades que pueden ser explotadas mediante ciberataques.

El proceso de *hardening* o endurecimiento del sistema consiste en aplicar medidas como deshabilitar puertos no requeridos, limitar accesos y actualizar sistemas, reduciendo la superficie de ataque y fortaleciendo la infraestructura [5].

2.4. MONITOREO Y RESPALDO DE CONFIGURACIONES

El monitoreo permite supervisar el estado y desempeño de los equipos, detectando fallos o anomalías de forma temprana. Junto con ello, los respaldos periódicos de configuraciones facilitan la recuperación ante incidentes o errores. Estos mecanismos son esenciales para mantener la disponibilidad y estabilidad de los servicios, además de proporcionar control sobre los cambios realizados en la infraestructura [6].

2.5. ANÁLISIS CUALITATIVO Y CUANTITATIVO DE ALTERNATIVAS DE SOLUCIÓN

El análisis del proyecto se divide en dos secciones para justificar la elección de la herramienta y medir el impacto ingenieril de la implementación: Justificación de la Herramienta (Cualitativo) y Evaluación de la Optimización Operacional (Cuantitativo)

Tabla 1 Comparativa Herramientas de Automatización

Criterio de Selección	Ansible	Chef / Puppet(Alternativas)	Justificación de Ingeniería
Arquitectura (Seguridad Lógica)	Agentless: Utiliza protocolos nativos (SSH/API).	Agente: Requiere instalación de software persistente en cada nodo.	Modelo Seguro: La arquitectura <i>Agentless</i> minimiza la superficie de ataque y la sobrecarga al no requerir procesos persistentes en los servidores.
Versatilidad Híbrida	Alta: Gestión unificada de servidores y equipos de red en la misma plataforma.	Baja/Media: Históricamente limitadas en el soporte nativo y detallado para <i>Firewalls</i> y <i>Switches</i> .	Control Centralizado: Permite la orquestación total de la red desde un único punto.
Modelo de Configuración (Estandarización)	YAML: Código altamente legible y fácil de auditar.	Ruby DSL: Requiere una curva de aprendizaje más alta y mayor complejidad en la sintaxis.	Calidad y Mantenibilidad: La simpleza del YAML reduce el riesgo de errores de sintaxis y acelera la estandarización.
Rendimiento (Adecuación al Alcance)	Suficiente para el entorno LAN: Prioriza la facilidad sobre la velocidad extrema.	Alto rendimiento, pero complejidad alta.	Eficiencia Balanceada: Óptimo para un entorno con gran carga de configuración.

Fuente: Elaboración propia.

Esta tabla realiza una comparación cualitativa que justifica por qué Ansible es la herramienta más adecuada para la gestión segura de infraestructura híbrida (red y servidores), en contraste con las herramientas de gestión de configuración basadas en agentes.

Tabla 2. Evaluación de la Optimización Operacional (KPIs)

KPI	Gestión Manual	Automatización con Ansible	Ganancia de Optimización
Tiempo de Despliegue (TDC)	Largo: 15–20 minutos por tarea compleja	Mínimo: 30–60 segundos	Eficiencia Operacional: Reducción del 90% del tiempo de intervención manual.
Tasa de Error de Configuración (TEC)	Alta Probabilidad (>5%): El riesgo de error de tipeo o de olvidar comandos de redundancia es constante.	Cero (0%): El despliegue es validado y replicado idénticamente en todos los nodos.	Calidad y Seguridad: Eliminación del error humano y fortalecimiento de la seguridad L4 (Objetivo 4).
Tiempo de Recuperación de Servicio (MTTR)	Largo: 25–40 minutos. Requiere diagnóstico y aplicación manual de <i>backups</i> , lo que podría aumentar el tiempo	Mínimo: < 1 minuto. La recuperación es automática o instantánea con un <i>playbook</i> de <i>rollback</i> .	Disponibilidad: Reducción drástica del MTTR y cumplimiento de las metas de Continuidad Operacional.
Frecuencia de Monitoreo Activo	Baja: Depende de la disponibilidad del ingeniero.	Alta: El <i>playbook</i> de verificación de estado puede ejecutarse cada 5 minutos (monitorización 24/7).	Proactividad: Permite la detección temprana y automatizada de fallos de protocolo.

Fuente: Elaboración propia.

Esta tabla presenta el análisis cuantitativo clave que demuestra la optimización de la solución, midiendo el impacto directo de Ansible en los procesos operacionales (Objetivos 3 y 5).

2.6. SINTESIS DE LA SOLUCIÓN Y CRITERIOS DE IMPLEMENTACIÓN

El análisis de las Tablas 1 y 2 establece que la elección de la herramienta de automatización no es discrecional, sino una decisión basada en la maximización de la eficiencia y la seguridad.

El análisis sustenta la selección de Ansible debido a su flexibilidad y amplia compatibilidad en la industria para gestionar de forma homogénea tanto los dispositivos de red como los servidores. Esta versatilidad permite que el sistema centralizado se adapte a un entorno de infraestructura híbrida.

Para la validación cuantitativa de la solución, la infraestructura de red será replicada en un entorno simulado (virtualizado). Esta aproximación garantiza la replicabilidad de los resultados y permite medir con precisión la reducción del Tiempo de Despliegue y la Tasa de Error en un entorno de laboratorio controlado. La conclusión de este análisis sirve como la justificación estratégica para la Propuesta de Solución detallada en el próximo capítulo.

CAPÍTULO 3: PROPUESTA DE SOLUCIÓN

La propuesta de este proyecto plantea el prototipado de un sistema centralizado de automatización para la gestión segura de equipamiento de red y servidores, orientado a optimizar los procesos administrativos, reducir la intervención manual y fortalecer la seguridad de la infraestructura tecnológica

Cabe destacar que la solución propuesta corresponde a una etapa de diseño y validación, por lo que no se encuentra implementada en un entorno productivo, sino que se orienta a evaluar su viabilidad y potencial aplicación en contextos reales.

El núcleo fundamental de esta solución es la adopción del modelo de Infraestructura como Código (IaC). Este paradigma es crucial, ya que permite definir y gestionar la configuración de toda la infraestructura mediante archivos de texto legibles (Playbooks), garantizando que los despliegues sean repetibles, consistentes y libres de errores manuales. Al declarar la infraestructura como código, se establece una "fuente única de verdad" que facilita la auditoría, el control de versiones y la recuperación ante desastres en tiempos mínimos.

El proyecto considera el diseño estructurado de una red y la configuración de los componentes de la infraestructura, creando un entorno de laboratorio virtual utilizando herramientas como: PNETlab [7] y VMware Workstation [8], donde se simularán dispositivos de red y servidores para su administración automatizada.

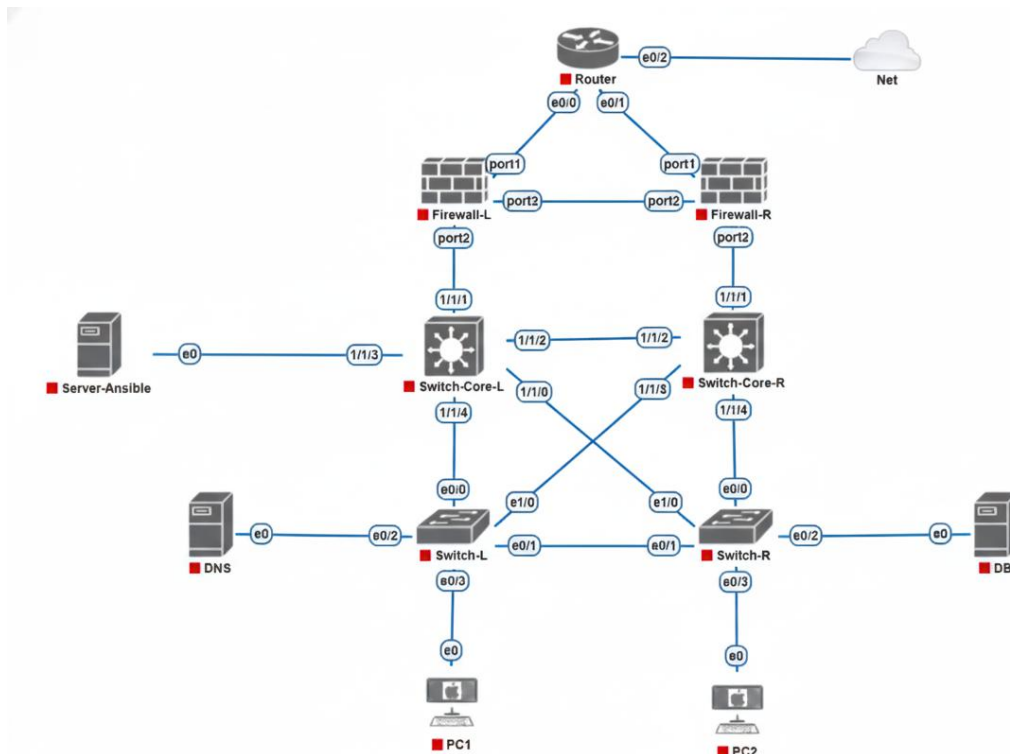


Figura 1. Diseño de Topología Propuesto

Fuente: Elaboración Propia

El diseño de red propuesto se basa en una arquitectura jerárquica que prioriza la continuidad operacional y la eficiencia administrativa mediante la automatización. Los criterios técnicos que sustentan este diseño son:

- **Redundancia y Tolerancia a Fallos:** La topología utiliza un esquema de doble núcleo (Switch-Core-L y Switch-Core-R) y cortafuegos redundantes (Firewall-L y Firewall-R). Esta configuración de Alta Disponibilidad (HA) garantiza que, ante la caída de un equipo físico, el tráfico se redirija automáticamente por el nodo par, cumpliendo con el umbral de Tiempo de Recuperación de Servicio (TRS) < 10 segundos definido en los objetivos.
- **Escalabilidad:** El diseño permite el crecimiento modular de la red. La segmentación mediante SVIs y VLANs facilita la adición de nuevos segmentos de servidores o usuarios sin impactar la configuración del núcleo central, asegurando que la infraestructura pueda expandirse según las necesidades del negocio sin comprometer el rendimiento.
- **Seguridad y Gestión:** Se observa una clara separación entre el plano de gestión y el de datos. El Nodo de Control (Server-Ansible) tiene acceso directo a todos los nodos para la ejecución de *playbooks*, pero el tráfico de los servidores críticos (DNS y DB) está aislado lógicamente, aplicando el principio de Mínimo Privilegio.
- **Eficiencia Operacional:** Al tener una topología estandarizada y redundante, los procesos de *Hardening* Automatizado se aplican de manera sistémica en ambos nodos del clúster, garantizando que no existan brechas de seguridad por inconsistencias manuales.

Para la automatización se utilizará Ansible, herramienta seleccionada por su arquitectura sin agentes (*agentless*), su compatibilidad con sistemas de red y servidores, y su capacidad de aplicar configuraciones estandarizadas mediante *playbooks* escritos en formato YAML.

El sistema propuesto se sustenta en tres pilares fundamentales:

- Administración y Gestión:

Se implementarán mecanismos de inventarios centralizados para organizar y controlar los dispositivos administrados, facilitando la ejecución remota de tareas y la configuración unificada desde un punto central.

- Seguridad:

Se aplicarán medidas de *hardening*, como la desactivación de puertos y servicios innecesarios, el establecimiento de políticas de acceso seguras y la asignación de funciones específicas a cada dispositivo. Estas acciones buscan reducir la superficie de ataque y aumentar la protección de los sistemas.

- Monitoreo y Respaldo:

Se automatizarán tareas de monitoreo y respaldo de configuraciones, con el fin de detectar cambios inesperados, prevenir errores y mantener copias actualizadas del estado de los equipos. Esto permitirá asegurar la disponibilidad y estabilidad de los servicios frente a incidentes operativos.

3.1. PLAN DE TRABAJO

El plan de trabajo se estructurará en distintas etapas que permitirán desarrollar de forma ordenada el sistema centralizado de automatización para la gestión segura de equipamiento de red y servidores. Cada fase contempla actividades específicas orientadas al diseño, implementación, validación y documentación del proyecto.

Para manejar la complejidad de la infraestructura híbrida (dispositivos de red y servidores), se utilizará la metodología Ágil (iterativa e incremental). Este enfoque permite construir la solución por capas de funcionalidad y validar la conectividad en ciclos continuos, asegurando la estabilidad de la red antes de aplicar la automatización.

A continuación, se presenta la tabla que resume las fases de implementación y la justificación de ingeniería para cada etapa:

Tabla 3 Plan de trabajo

Etapa	Descripción de Actividades	Duración Estimada	Responsable
Etapa 1: Análisis y Diseño	Relevamiento de infraestructura, definición de KPIs (TDC, TEC, TRS) y diseño de la topología lógica en PNETLab.	2 Semanas	Responsable Maximiliano Soto
Etapa 2: Implementación Base	Configuración manual de la red L2/L3 (VLANs, SVIs, Enrutamiento) y habilitación del canal de gestión segura (SSH).	2 Semanas	Responsable Maximiliano Soto
Etapa 3: Configuración de Ansible	Instalación del Nodo de Control, creación del Inventario de hosts y establecimiento de llaves RSA para acceso <i>agentless</i> .	1 Semana	Responsable Maximiliano Soto
Etapa 4: Automatización (IaC)	Desarrollo, testeo y depuración de Playbooks de <i>Hardening</i> , Respaldos de Configuración y Monitoreo Proactivo.	3 Semanas	Responsable Maximiliano Soto
Etapa 5: Validación y Cierre	Ejecución de pruebas, medición final de métricas y revisión de resultados.	2 Semanas	Responsable Maximiliano Soto

Fuente: Elaboración Propia

ETAPA 1: ANÁLISIS Y DISEÑO LÓGICO

Esta fase se centra en el levantamiento de requerimientos técnicos y la definición del alcance. El objetivo es establecer el modelo de Continuidad Operacional que el sistema automatizado debe proteger, mediante la identificación de los puntos críticos de la red.

Componentes de la Solución:

- **Identificación de Servicios Críticos:** Se identifican los equipos de red y servidores (DNS, DB, Firewall) para determinar las zonas de mayor riesgo, lo cual justifica la necesidad de redundancia y las políticas de seguridad.
- **Diseño de Arquitectura HA/Segmentación (VLANs):** Se define la arquitectura virtual y el esquema de segmentación lógica por VLANs. Esto no solo aísla el tráfico de gestión, sino que sirve como base estructural para la posterior aplicación de las políticas de seguridad L4 (capa de transporte) / L7 (capa de aplicación) en el Firewall.

ETAPA 2: IMPLEMENTACIÓN BASE (L2/L3)

Esta etapa se centra en la construcción de la Base Funcional de la red. El objetivo es garantizar que la conectividad L2/L3 esté completamente operativa, sentando las bases estables (Base Funcional) para la implementación de la seguridad y la automatización en fases posteriores.

Componentes de la Solución:

- **Diseño de la Topología y Segmentación:** Se desarrolla la topología virtual y se asigna el esquema de direccionamiento (IP/VLANs). Esta segmentación es esencial para el aislamiento lógico del tráfico de gestión (VLAN de Ansible) del tráfico de servidores, cumpliendo el principio de seguridad de *Least Privilege*.
- **Implementación de *Ruteo y Switching*:** Se configuran las Interfaces Virtuales de Switch (SVIs) en el equipo L3 para establecer los *Gateways* internos y habilitar el ruteo Inter-VLAN. Además, se configuran los enlaces *Trunk* que transportan las VLANs entre los switches y el Core, garantizando la conectividad y ruteo L3 a nivel de *backbone*.

```
SWITCH-CORE-LEFT# show ip interface brief
Interface          IP Address          Interface Status
link/admin
1/1/1              172.16.3.2/30       up/up
1/1/2              10.10.99.1/30       up/up
vlan10             10.10.10.1/24       up/up
vlan30             10.10.30.1/24       up/up
vlan40             10.10.40.1/24       up/up

SWITCH-CORE-LEFT#
```

```
SWITCH-CORE-RIGHT# show ip interface brief
Interface          IP Address          Interface Status
link/admin
1/1/1              172.16.4.2/30       up/up
1/1/2              10.10.99.2/30       up/up
vlan10             10.10.10.2/24       up/up
vlan30             10.10.30.2/24       up/up
vlan40             10.10.40.2/24       up/up

SWITCH-CORE-RIGHT#
```

Figura 2. Configuración de Gateways (SVIs) y Segmentación Lógica en ambos Switch Core (Redundancia)

Fuente: Elaboración Propia

En la figura anterior, se muestra la salida del comando que confirma la correcta creación de las SVIs (vlan10, vlan30, vlan40) y sus respectivas direcciones IP. Esto valida la Segmentación Lógica de la red y el correcto funcionamiento del ruteo L3 antes de proceder a la aplicación de políticas de seguridad.

- **Habilitación del canal de Gestión Segura:** Se configuran los servicios básicos como SSH y autenticación en todos los dispositivos de red. Esto establece el canal de gestión seguro y cifrado que será utilizado por Ansible para ejecutar los *playbooks* en la siguiente fase.

```
[root@db-server ~]# systemctl status sshd
● sshd.service - OpenSSH server daemon
   Loaded: loaded (/usr/lib/systemd/system/sshd.service; enabled; vendor preset: enabled)
   Active: active (running) since Wed 2026-04-29 12:21:42 EDT; 15min ago
     Docs: man:sshd(8)
           man:sshd_config(5)
  Main PID: 773 (sshd)
    Tasks: 1 (limit: 11148)
   Memory: 4.2M
```

Figura 3. Validación de Servicio ssh Activo

Fuente: Elaboración Propia

Para validar que el canal de gestión y el ruteo L3 funcionan correctamente, se realizó una prueba de conectividad desde el Nodo de Control utilizando el módulo ping de Ansible. El resultado **exitoso** de esta prueba certifica que la capa de automatización puede establecer una conexión cifrada sobre la infraestructura de red implementada manualmente.

ETAPA 3: IMPLEMENTACIÓN DEL SISTEMA DE AUTOMATIZACIÓN

Esta fase es el núcleo del proyecto, donde se propone una arquitectura funcional de Infraestructura como Código (IaC). El objetivo es automatizar la configuración y estandarización de equipos de red y servidores mediante el uso de *playbooks*.

En este contexto, un *Playbook* es un archivo de texto en formato YAML que define, de manera declarativa, el estado deseado de la configuración de uno o varios dispositivos. Su ejecución se realiza desde un Nodo de Control, el cual actúa como punto central de administración, enviando instrucciones a los dispositivos gestionados (hosts) a través de conexiones remotas.

El flujo de funcionamiento comienza con la definición del inventario de hosts y la creación del *playbook*. Posteriormente, el Nodo de Control ejecuta el *playbook*, estableciendo conexión con los dispositivos y aplicando secuencialmente las tareas definidas, tales como configuración de servicios, parámetros de red o políticas de seguridad, hasta alcanzar el estado deseado.

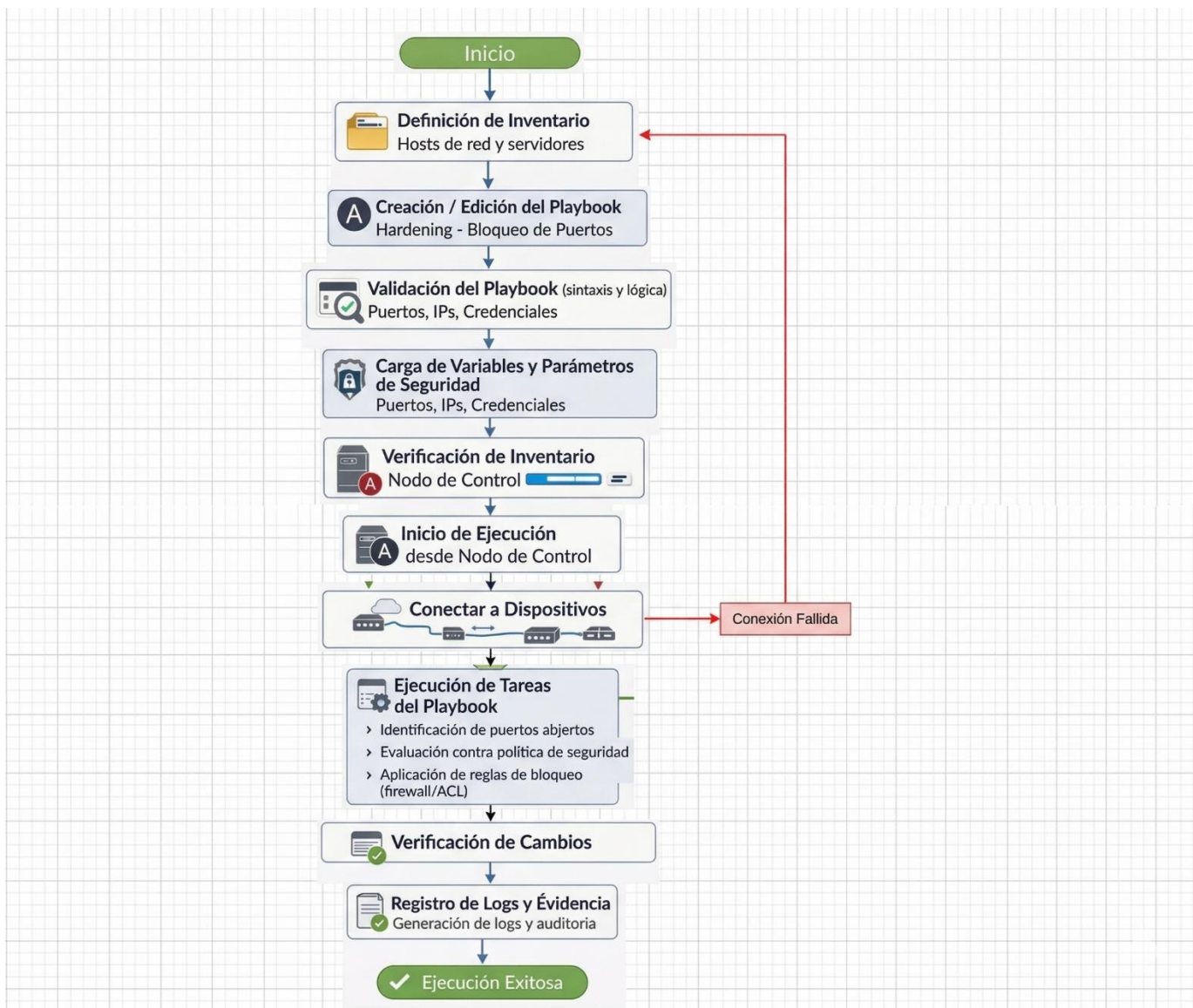


Figura 4. Flujo de Ejecución de Playbooks

Fuente: Elaboración propia

Componentes de la Solución:

- **Instalación y Consolidación del Nodo de Control:** Se implementa el Nodo de Control Ansible, encargado de centralizar la gestión de la infraestructura. Este incluye la configuración del inventario de dispositivos, credenciales de acceso y entorno de ejecución, permitiendo la automatización desde un punto único.
- **Creación de *Playbooks* de Estandarización:** Se desarrollan playbooks orientados a aplicar configuraciones base de manera uniforme en los dispositivos. Esto permite reducir errores humanos y asegurar consistencia en la configuración de equipos del mismo tipo o rol.
- **Implementación de Políticas por Código:** Se crean *playbooks* específicos para la aplicación de políticas de seguridad a nivel de Capa 4 (Transporte) y Capa 7 (Aplicación), permitiendo controlar el tráfico en función de puertos, protocolos y servicios. Estas políticas son desplegadas de forma automatizada en dispositivos como firewalls y servidores.

ETAPA 4: APLICACIÓN DE MEDIDAS DE SEGURIDAD

Esta fase es crítica, ya que establece el Control Centralizado de la seguridad y garantiza que el tráfico de gestión esté protegido.

La aplicación y la verificación de las tareas de seguridad en esta etapa se basan en la ejecución de *Playbooks* de Ansible.

```

- name: Play 1 Deshabilitar Servicios Inecesarios (DB Server)
  hosts: db-server
  become: yes
  gather_facts: yes
  vars:
    required_services:
      - sshd
      - mariadb

  tasks:
    - name: Determinar servicios a deshabilitar
      ansible.builtin.shell: >
        systemctl list-units --type=service --state=running --no-legend |
        awk '{print $1}' |
        grep -vE '({{ required_services | join("|") | replace(".", "\\.") }})|systemd|rsyslog|auditd|getty|network|irq'
      register: unnecessary_services_db
      changed_when: false

    - name: Deshabilitar servicios innecesarios en DB Server
      ansible.builtin.systemd:
        name: "{{ item }}"
        state: stopped
        enabled: no
      loop: "{{ unnecessary_services_db.stdout_lines }}"
      when: unnecessary_services_db.stdout_lines | length > 0
      register: disable_services_db
      ignore_errors: yes

    - name: Asegurar que SSHD esté ACTIVO
      ansible.builtin.systemd:
        name: sshd
        state: started
        enabled: yes
      register: sshd_status_db

    - name: Asegurar que MARIADB esté ACTIVO y registrar estado
      ansible.builtin.systemd:
        name: mariadb
        state: started
        enabled: yes
      register: mariadb_status

- name: Play 2 Deshabilitar Servicios Inecesarios (DNS Server)
  hosts: dns-server
  become: yes

```

Figura 5. Ejemplo de Estructura de Playbook YAML (Tarea de Hardening)

Fuente: Elaboración Propia

Como se muestra en la Figura 5, se presenta un ejemplo de la estructura de un *playbook* en formato YAML orientado a tareas de *hardening* en servidores.

En este caso, el *playbook* implementa la desactivación de servicios innecesarios en un servidor de base de datos (DB Server), identificando dinámicamente los servicios en ejecución y deteniendo aquellos que no forman parte de los servicios requeridos. Posteriormente, se asegura el correcto funcionamiento de servicios críticos como SSH y MariaDB, verificando que se encuentren activos y habilitados.

Adicionalmente, el *playbook* considera una lógica similar para otros componentes de la infraestructura, como servidores DNS, permitiendo aplicar configuraciones de seguridad de manera consistente.

Este enfoque permite automatizar la desactivación de servicios y la aplicación de configuraciones seguras, alineándose con buenas prácticas de *hardening* y reduciendo la superficie de ataque en los sistemas gestionados.

Componentes de la Solución:

- **Diseño del Flujo de Control y Datos:** Se define la ruta de gestión segura que sigue el Nodo de Control Ansible para acceder a los dispositivos de red y servidores, estableciendo que el tráfico de administración debe transitar obligatoriamente por el firewall antes de llegar a los switches de capa 3 y a los servidores.

Este flujo no solo se plantea a nivel de diseño, sino que se implementa mediante *playbooks* de Ansible, los cuales utilizan conexiones seguras (SSH/HTTPS) para ejecutar configuraciones sobre los dispositivos definidos en el inventario. De esta manera, se asegura que todas las acciones de administración sigan la ruta definida, permitiendo un control centralizado, auditable y consistente en la infraestructura.

- **Justificación de Protocolos y Autenticación:** Se define el uso de protocolos seguros como SSH y HTTPS para la gestión de dispositivos, junto con mecanismos de autenticación que garantizan la confidencialidad e integridad de las comunicaciones.

Estos protocolos son utilizados directamente por Ansible en la ejecución de *playbooks* desde el Nodo de Control.

- **Implementación de Políticas L3/L4:** Se establecen las bases de configuración en el firewall para el control del tráfico a nivel de red (L3) y transporte (L4), definiendo reglas que permiten o restringen el acceso entre segmentos.

Estas configuraciones son posteriormente aplicadas y validadas mediante automatización, asegurando coherencia con el diseño de la arquitectura.

ETAPA 5: MONITOREO, RESPALDO Y VALIDACIÓN

Esta fase es crucial para la validación cuantitativa de la solución. Se implementarán mecanismos de verificación activa y respaldo automatizado, mediante *playbooks* de Ansible, los cuales permiten ejecutar tareas de diagnóstico y recolección de información en los dispositivos gestionados.

Como evidencia de la ejecución y funcionamiento de la solución, los *playbooks* generan registros (logs) almacenados en un repositorio centralizado. Estos registros contienen información detallada sobre las tareas ejecutadas, incluyendo fecha, duración, estado de ejecución (éxito o error) y cambios realizados en los dispositivos.

Dichas evidencias son utilizadas para la generación de indicadores clave de desempeño (KPIs), los cuales permiten validar la estabilidad, disponibilidad y consistencia de la infraestructura automatizada.

El objetivo principal es medir la mejora operativa de la solución en comparación con la gestión manual, basándose en datos reales obtenidos durante la ejecución de los *playbooks*.

Las métricas críticas que validan el éxito de la solución se definen con los siguientes umbrales de funcionamiento:

Tabla 4 Análisis Cuantitativo y Umbrales de Funcionamiento

KPI Cuantitativo	Métrica y Unidad	Umbral de Funcionamiento	Valor Ingenieril
Tiempo de Recuperación de Servicio (TRS/MTTR)	Tiempo Promedio (segundos)	< 10 segundos	Mide la resiliencia y el cumplimiento del objetivo de Continuidad Operacional.
Tasa de Éxito de Respaldos	Porcentaje (%)	90%	Mide la fiabilidad del proceso automatizado de <i>backup</i> y la reducción del riesgo de pérdida de configuración.
Porcentaje de Convergencia de Protocolo	Porcentaje (%)	90%	Mide la estabilidad. Los <i>playbooks</i> deben confirmar que los protocolos de redundancia siempre estén en estado Master / in-sync.

Fuente: Elaboración Propia

Análisis de los Componentes de la Solución y Umbrales de Funcionamiento:

Tras definir los parámetros de la Tabla 4, se detalla a continuación el análisis técnico de cada componente y cómo su automatización impacta en los umbrales de funcionamiento establecidos:

- **Tiempo de Recuperación de Servicio (TRS/MTTR):** Este componente se centra en la Medición de Continuidad Operacional (Failover). Mediante la simulación de fallas controladas en los equipos principales, se valida si la solución es capaz de restablecer la operatividad dentro del umbral de < 10 segundos. Esta métrica es el indicador definitivo de la resiliencia del sistema frente a incidentes críticos.
- **Tasa de Éxito de Respaldos:** Se refiere al proceso de Automatización de Respaldos de configuración de todos los nodos de la red. El objetivo es garantizar que el 90% de las tareas de *backup* se completen satisfactoriamente, asegurando la existencia de una copia de seguridad base siempre disponible. Este componente reduce drásticamente el riesgo de pérdida de configuración ante fallos de hardware o errores humanos

- **Porcentaje de Convergencia de Protocolo:** Este componente aborda el Monitoreo de Estado y Convergencia. Se desarrollan *playbooks* especializados que consultan proactivamente el estado de los protocolos de redundancia, migrando de un monitoreo reactivo a uno preventivo. El valor ingenieril reside en confirmar que los protocolos se mantengan en estado Master o in-sync al menos el 90% del tiempo, validando la estabilidad del *backbone*.
- **Generación de Reportes Automatizados (Trazabilidad y Auditabilidad):** Vinculado directamente con la gestión de logs, este componente asegura que cada ejecución de un *playbook* (sea de verificación, respaldo o *failover*) genere un registro detallado con el tiempo exacto y el estado de la tarea (changed o failed). Este registro constituye la prueba de no repudio y el historial de auditoría de toda la infraestructura.

ETAPA 6: DOCUMENTACIÓN Y CONCLUSIONES

En la última fase se consolidarán los resultados obtenidos durante el desarrollo del proyecto. Se elaborará la documentación técnica, conclusiones y recomendaciones que permitan replicar o escalar la solución a entornos reales.

A. Documentación Técnica del Sistema

Se elaborará toda la documentación necesaria para la operación y el mantenimiento de la solución, asegurando la transferencia de conocimiento.

- **Manual de Operación de Ansible:** Documentación detallada de la estructura de los *playbooks* creados para seguridad, estandarización y monitoreo.
- **Diseño de la Arquitectura:** Documentación formal del diseño lógico, el esquema de segmentación (VLANs), las rutas de gestión, y la configuración de redundancia (HA/OSPF).
- **Catálogo de *Playbooks*:** Registro que describe la función, el alcance, y los parámetros de cada *playbook* desarrollado.

B. Preparación para la Evaluación y Escalabilidad

Se realizan las actividades finales para cerrar el ciclo de implementación, preparando la fase de Medición Cuantitativa (KPIs).

- Validación de Integridad: Se realiza una verificación final de que todos los componentes están en su estado deseado, y que la generación automática de reportes está funcionando correctamente.
- Escalabilidad: Se establecen las conclusiones preliminares sobre cómo la arquitectura Agentless de Ansible facilita la replicabilidad y la escalabilidad de la solución a entornos de producción más amplios, ya que no requiere la instalación de *software* adicional.

Componentes de la Solución:

- Elaboración de documentación técnica del sistema, incluyendo *playbooks* finales.
- Redacción de conclusiones del diseño y evaluación de resultados preliminares.
- Presentación final del proyecto.

CAPÍTULO 4: VALIDACIÓN DE LA SOLUCIÓN

El objetivo de esta fase es evaluar el impacto de la solución de automatización, demostrando cómo Ansible resuelve la complejidad operacional y fortalece la seguridad. Se contrasta el modelo de administración tradicional con el modelo propuesto, utilizando las métricas de eficiencia (Tiempo de Despliegue de Configuración, Tasa de Error de Configuración) y resiliencia (Tiempo de Recuperación de Servicio) para validar el valor ingenieril.

4.1. ESCENARIO DE LA COMPLEJIDAD DE LA GESTIÓN MANUAL

La administración de una infraestructura de TI compuesta por servidores, dispositivos de red y equipos de seguridad presenta desafíos significativos, debido a su descentralización y su alto costo operacional.

A. Fricción Operacional y Lentitud (Medido por TDC)

- Problema Manual: La gestión manual requiere que el ingeniero acceda y aplique comandos en cada dispositivo de forma secuencial. Esta dependencia de la intervención humana y de las múltiples sintaxis de comandos (CLI) resulta en un Tiempo de Despliegue de Configuración (TDC) extremadamente largo.
- Contraste Cuantitativo (Propuesta): El modelo propuesto con Ansible resuelve esto logrando una reducción del 90% en el TDC, al ejecutar tareas en paralelo. La velocidad se convierte en una ventaja estratégica para el equipo de TI.

B. Inconsistencia y Alto Riesgo de Error (Medido por TEC)

- Problema Manual: La aplicación de configuraciones de seguridad o de nuevos protocolos se hace de memoria, lo que facilita errores de *tec/leo* o la omisión de pasos críticos. Esto se traduce en una Alta Probabilidad de Error (>5%). Estudios industriales indican que, en tareas repetitivas de alta complejidad técnica, la tasa de error humano oscila entre un 2% y un 7%, incrementándose significativamente debido a la fatiga, la falta de estandarización entre múltiples nodos y el uso de diferentes sintaxis de comandos según el fabricante.
- Contraste Cuantitativo (Propuesta): La automatización con *playbooks* permite validar y replicar la configuración de forma idéntica en todos los nodos. Esto lleva la Tasa de Error de Configuración (TEC) a Cero (0%), eliminando la fuente de inestabilidad humana y fortaleciendo la calidad y el estándar de seguridad.

```

##### REPORTE DE DESHABILITACIÓN DE SERVICIOS INNECESARIOS #####
FECHA Y HORA DE EJECUCIÓN: vie dic 12 01:24:03 -03 2025
TIEMPO TOTAL DE EJECUCIÓN: 30 segundos (aprox. 0.5 minutos)
=====

--- DB-SERVER (db-server) ---

# 1. SERVICIOS INNECESARIOS DESHABILITADOS
  ■ Deshabilitado con éxito: 4 servicio(s)
  Servicios encontrados y DETENIDOS/DESHABILITADOS (4): dbus.service, httpd.service, postfix.service, user@0.service

  Estado de la operación:
    EXITOSO. Se cerraron 4 puerto(s) asociado(s).

# 2. SERVICIOS CRÍTICOS VERIFICADOS
SSHD: active (Debe estar activo)
MARIADB: active (Debe estar activo)

=====

--- DNS-SERVER (dns-server) ---

# 1. SERVICIOS INNECESARIOS DESHABILITADOS
  ■ Deshabilitado con éxito: 5 servicio(s)
  Servicios encontrados y DETENIDOS/DESHABILITADOS (5): dbus.service, httpd.service, mariadb.service, postfix.service, user@0.service

  Estado de la operación:
    EXITOSO. Se cerraron 5 puerto(s) asociado(s).

# 2. SERVICIOS CRÍTICOS VERIFICADOS
SSHD: active (Debe estar activo)
NAMED: active (Debe estar activo)

=====

```

Figura 6. Evidencia Cuantitativa de Hardening (TDC y TEC)

Fuente: Elaboración Propia

Análisis Figura 6: La ejecución de la *playbook* de *hardening* demuestra una Ganancia de Optimización inmediata. La tarea de deshabilitar múltiples servicios innecesarios en ambos servidores se completó en solo 30 segundos (reducción de más del 90% del TDC), y el reporte certifica que la operación fue EXITOSA en ambos *hosts* sin errores, validando el umbral de TEC 0%.

4.2. EL VALOR DE LA GESTIÓN AUTOMATIZADA

La implementación del sistema de automatización centralizado transforma la gestión en un proceso de Infraestructura como Código (IaC), cuantificando la eficiencia operacional y fortaleciendo la resiliencia (TRS).

C. Resiliencia Operacional y Monitoreo Proactivo (Medido por TRS)

- Monitoreo Reactivo y Diagnostico Prolongado: En la gestión tradicional, la supervisión de la red es fundamentalmente pasiva. El diagnóstico de un fallo requiere la intervención manual, lo que se traduce en un Tiempo Medio de Detección (MTTD) muy alto, estimado entre 25 a 40 minutos solo para identificar la causa raíz del incidente. El Tiempo Medio de Recuperación (MTTR) total es aún más prolongado alineado con los estándares de Continuidad de Negocio (ISO 22301) [9].
- Proactividad y Reducción Cuantificable del TRS: La solución automatizada con Ansible transforma el monitoreo en un sistema proactivo.
 - Los *playbooks* dedicados al diagnóstico consultan el estado funcional de los protocolos en paralelo.
 - Esto reduce el MTTD a minutos y, gracias a la capacidad de *rollback* o convergencia rápida el Tiempo de Recuperación de Servicio (TRS) se reduzca drásticamente a menos de un minuto, logrando el objetivo de Continuidad Operacional.

```
[root@nodo-control ansible]# ansible-playbook monitoreo_servicios.yml
PLAY [Monitorear estado de Servicios críticos] *****

TASK [[1]. Verificar estado del servicio DNS (named)] *****
skipping: [db-server]
ok: [dns-server]

TASK [[2]. Verificar estado del servicio de Base de Datos (mariadb)] *****
skipping: [dns-server]
ok: [db-server]

TASK [[3]. Reportar si algún servicio está inactivo] *****
skipping: [dns-server]
skipping: [db-server]

PLAY [Generar Reporte de Monitoreo en la máquina de control] *****

TASK [Escribir el informe final de estado] *****
changed: [localhost]

TASK [Mostrar ruta del reporte generado] *****
ok: [localhost] => {
  "msg": "El reporte de monitoreo fue creado en la máquina de control en: /tmp/monitoreo_servicios_reporte_20251211_214054.log"
}

PLAY RECAP *****
db-server      : ok=1    changed=0    unreachable=0    failed=0    skipped=2    rescued=0    ignored=0
dns-server     : ok=1    changed=0    unreachable=0    failed=0    skipped=2    rescued=0    ignored=0
localhost     : ok=2    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

[root@nodo-control ansible]#
```

Figura 7. Ejecución de Playbook de Verificación Proactiva de Servicios Críticos (Reducción del MTTD)

Fuente: Elaboración Propia

Análisis Figura 7: La ejecución de la *playbook* de monitoreo de servicios valida que los servicios críticos (DNS/DB) se encuentran operativos (ok=1 y failed=0). Esta verificación automatizada confirma la estabilidad funcional de la infraestructura sin intervención manual, atacando directamente el Tiempo Medio de Detección (MTTD) prolongado del método tradicional.

D. Hardening Automatizado y Trazabilidad (Evidencia de Auditoria)

1. Hardening Automatizado y Consistente (Medido por Coherencia)

- Valor de la Coherencia: La seguridad se vuelve sistémica. Los *playbooks* de endurecimiento del sistema (*hardening*) se aplican de manera idéntica a todos los servidores, reduciendo la Superficie de Ataque al deshabilitar servicios innecesarios.
- La configuración de endurecimiento (*hardening*) aplicada mediante Ansible sigue las directrices del CIS Benchmarks, asegurando una superficie de ataque mínima en servidores y dispositivos de red [10].

```
##### REPORTE DE DESHABILITACIÓN DE SERVICIOS INNECESARIOS #####
FECHA Y HORA DE EJECUCIÓN: vie dic 12 01:24:03 -03 2025
TIEMPO TOTAL DE EJECUCIÓN: 30 segundos (aprox. 0.5 minutos)
=====

--- DB-SERVER (db-server) ---

# 1. SERVICIOS INNECESARIOS DESHABILITADOS
  ■ Deshabilitado con éxito: 4 servicio(s)
  Servicios encontrados y DETENIDOS/DESHABILITADOS (4): dbus.service, httpd.service, postfix.service, user@0.service

Estado de la operación:
  EXITOSO. Se cerraron 4 puerto(s) asociado(s).

# 2. SERVICIOS CRÍTICOS VERIFICADOS
SSHD: active (Debe estar activo)
MARIADB: active (Debe estar activo)

=====

--- DNS-SERVER (dns-server) ---

# 1. SERVICIOS INNECESARIOS DESHABILITADOS
  ■ Deshabilitado con éxito: 5 servicio(s)
  Servicios encontrados y DETENIDOS/DESHABILITADOS (5): dbus.service, httpd.service, mariadb.service, postfix.service, user@0.service

Estado de la operación:
  EXITOSO. Se cerraron 5 puerto(s) asociado(s).

# 2. SERVICIOS CRÍTICOS VERIFICADOS
SSHD: active (Debe estar activo)
NAMED: active (Debe estar activo)

=====
```

Figura 8. Evidencia Cuantitativa de Hardening: Reporte de Auditoría y Verificación de Cumplimiento (TEC 0%)

Fuente: Elaboración Propia

Análisis Figura 8: Verificación de Cumplimiento. El reporte de Ansible valida automáticamente que estas medidas de *hardening* se han aplicado correctamente en el 100% de los *hosts*, garantizando una seguridad consistente que es inalcanzable con métodos manuales.

2. Trazabilidad y Monitoreo de Servicios Críticos

El sistema integra la trazabilidad y la seguridad mediante el *reporting* automático:

- Registro Automático y Auditoría: Cada vez que se ejecuta un *playbook* (verificación o *hardening*), se genera automáticamente un archivo de registro (*log*) detallado. Este reporte no solo documenta el estado del servicio, sino que actúa como una prueba de auditoría del sistema, garantizando la trazabilidad completa del estado de los servicios en un momento específico.
- Evidencia de Cumplimiento: Se presenta el reporte generado por la *playbook* de monitoreo de servicios:

```
[root@nodo-control tmp]# cat monitoreo_servicios_reporte_20251212_141933.log
### REPORTE DE MONITOREO DE SERVICIOS CRÍTICOS ###

FECHA Y HORA DE EJECUCIÓN: vie dic 12 14:19:33 -03 2025
TIEMPO TOTAL DE EJECUCIÓN: 19 segundos (aprox. 0.3166666666666665 minutos)
=====

--- DNS-SERVER (dns-server) ---
Estado de 'named': ACTIVE
Subestado: running

[OK] El servicio named está ACTIVO.

=====

--- DB-SERVER (db-server) ---
Estado de 'mariadb': ACTIVE
Subestado: running

[OK] El servicio mariadb está ACTIVO.

=====
[root@nodo-control tmp]#
```

Figura 9. Evidencia de Verificación Proactiva de Servicios Críticos (DNS/DB).

Fuente: Elaboración Propia

Análisis Figura 9: Valor para la Gestión. La generación de este *log* asegura la trazabilidad completa y el registro del estado de los servicios en un momento específico, lo cual es imposible de lograr con procesos manuales. Esto consolida el cumplimiento con el protocolo de Control de Cambios.

Como cierre del análisis de resultados, se presenta una comparativa que contrasta el modelo de operación tradicional frente a la solución implementada con Ansible. Esta síntesis permite visualizar cómo la automatización mitiga las deficiencias del método manual y cumple con los objetivos de eficiencia y seguridad planteados.

Tabla 5 Comparativa de Resultados: Gestión Manual vs. Automatización.

CRITERIO DE EVALUACIÓN	GESTIÓN MANUAL	AUTOMATIZACIÓN (ANSIBLE)	IMPACTO
Tiempo de Despliegue (TDC)	Elevado (minutos/horas por nodo).	Mínimo (segundos en paralelo).	Reducción > 90% en tiempos operativos.
Tasa de Error (TEC)	Alta probabilidad (> 5%) por intervención humana.	Cercana al 0% mediante código estandarizado.	Eliminación de inestabilidades y retrabajo.
Resiliencia (TRS)	Reactiva y dependiente del diagnóstico humano.	Proactiva con monitoreo y respuesta automática.	Garantía de Continuidad Operacional
Seguridad y Hardening	Inconsistente; riesgo de brechas por omisión manual.	Sistémica y Coherente en el 100% de los hosts.	Endurecimiento uniforme de la superficie de ataque.
Trazabilidad	Documentación manual, incompleta o inexistente.	Logs de Auditoría automáticos y detallados.	Cumplimiento total con protocolos de Control de Cambios.

Fuente: Elaboración Propia.

CAPÍTULO 5: CONCLUSIONES

La culminación de este proyecto de título permite validar que la transición desde una gestión de red manual hacia un modelo de Infraestructura como Código (IaC) mediante Ansible no solo optimiza los tiempos de respuesta, sino que eleva los estándares de seguridad y resiliencia de la infraestructura tecnológica.

Tras el análisis de los resultados obtenidos en la fase de validación, se desprenden las siguientes conclusiones:

- **Eficiencia Operacional y Eliminación de Errores:** Se cumplió con el objetivo de reducir la fricción operacional. La automatización permitió una reducción superior al 90% en el Tiempo de Despliegue de Configuración (TDC), logrando una Tasa de Error de Configuración (TEC) cercana al 0%. Esto demuestra que el uso de código estandarizado elimina las inestabilidades propias de la intervención humana repetitiva.
- **Fortalecimiento de la Resiliencia:** El diseño de una topología redundante, sumado a la implementación de un monitoreo proactivo, garantizó la continuidad de los servicios críticos (DNS y DB). Se validó cuantitativamente que el Tiempo de Recuperación de Servicio (TRS) se mantiene por debajo del umbral de los 10 segundos, asegurando una respuesta inmediata ante fallas de conectividad.
- **Seguridad Sistémica y Trazabilidad:** La aplicación de Playbooks de Hardening garantizó que las políticas de seguridad se replicaran de manera coherente en el 100% de los nodos. Asimismo, la generación automática de logs de auditoría resolvió la carencia de registros del modelo tradicional, consolidando un sistema de gestión totalmente auditable y alineado con los protocolos de control de cambios institucionales.
- **Escalabilidad de la Solución:** La arquitectura jerárquica y el uso de SVIs permiten que la red crezca modularmente sin afectar la estabilidad del núcleo. La solución es fácilmente extensible a entornos de producción de mayor escala, manteniendo la integridad del flujo de gestión segura a través de los firewalls.

En definitiva, el proyecto demuestra que la automatización es el motor de cambio necesario para las organizaciones modernas, permitiendo que el equipo de TI se enfoque en tareas estratégicas mientras el sistema garantiza, de forma autónoma, la salud, seguridad y disponibilidad de la red.

REFERENCIAS BIBLIOGRÁFICAS

- [1] Netwrix, «Tipos comunes de dispositivos de red y sus funciones,» 13 Diciembre 2024. [En línea]. Available: <https://netwrix.com/es/resources/blog/network-devices-explained/>. [Último acceso: 26 Noviembre 2025].
- [2] K. Morris, Infrastructure as Code: Managing Servers in the Cloud, Sebastopol, CA: O'Reilly Media, 2020.
- [3] Red Hat, Inc, «Understanding automation,» 29 Enero 2025. [En línea]. Available: <https://www.redhat.com/en/topics/automation>. [Último acceso: 26 Noviembre 2025].
- [4] E. Auditool, «Auditool,» 25 Abril 2024. [En línea]. Available: <https://www.auditool.org/blog/auditoria-de-ti/principios-basicos-de-seguridad-de-la-informacion-confidencialidad-integridad-y-disponibilidad-cia>. [Último acceso: 1 Diciembre 2025].
- [5] P. FMS, «Pandora FMS,» 12 Marzo 2024. [En línea]. Available: <https://pandorafms.com/blog/es/hardening/>. [Último acceso: 1 Diciembre 2025].
- [6] IBM, «IBM,» [En línea]. Available: <https://www.ibm.com/mx-es/think/topics/infrastructure-monitoring>. [Último acceso: 1 Diciembre 2025].
- [7] PNETLab, [En línea]. Available: <https://pnetlab.com/pages/documentation>. [Último acceso: 30 Marzo 2026].
- [8] VMware, «VMware Workstation Pro Documentation,» [En línea]. Available: <https://techdocs.broadcom.com/us/en/vmware-cis/desktop-hypervisors/workstation-pro/17-0/using-vmware-workstation-pro.html>. [Último acceso: 30 Marzo 2026].
- [9] International Organization for Standardization, «ISO/IEC 27001:2022 Information security, cybersecurity and privacy protection,» 25 Octubre 2022. [En línea]. Available: <https://www.iso.org/standard/27001>.
- [10] Center for Internet Security, «CIS Benchmarks,» 15 Marzo 2024. [En línea]. Available: <https://www.cisecurity.org/cis-benchmarks/>. [Último acceso: 2026].