PONTIFICIA UNIVERSIDAD CATÓLICA DEL ECUADOR

FACULTAD DE INGENIERÍA

MAESTRÍA EN TECNOLOGÍAS DE LA INFORMACIÓN
MENCIÓN REDES DE COMUNICACIONES

TEMA:

DISEÑO Y SIMULACIÓN DE UNA RED LOCAL CONECTADA A PROVEEDORES DE SERVICIOS DE INTERNET (ISP) CON POLÍTICAS DE SEGURIDAD Y AUTOMATIZACIÓN DE SUS CONFIGURACIONES

AUTOR:

MAYRA ELIZABETH CAISATOA CHULCA

DIRECTOR:

ING. WLADIMIR EDISON MORALES OCANA MsC.

QUITO, 2023

ÍNDICE GENERAL

ÍNDICE DE FIGURAS

ÍNDICE DE TABLAS

ÍNDICE DE ANEXOS

RESUMEN

Este trabajo de titulación, tuvo como propósito el diseño y simulación de una red local conectada a proveedores de servicios de Internet que contiene políticas de seguridad implementadas en el dispositivo de Firewall ASA, y con programación para la configuración automatizada mediante la plataforma de código abierto Ansible. Esta red permite realizar configuraciones automáticas de las interfaces, protocolo de enrutamiento MPLS y OSPF, políticas de seguridad.

El capítulo uno, refiere a la introducción de la tesis, en donde incluye el planteamiento del problema, justificación y los objetivos para la fabricación de este documento. En el segundo capítulo, se tiene la revisión de la literatura y marco teórico, necesaria para la implementación de la red local conectada a proveedor ISP que contiene políticas de seguridad, configurado de manera automática.

En el capítulo tres, se encuentra la metodología de investigación, en donde se especifica el diseño y modalidad de investigación que se implementó en el trabajo de titulación, además, se realiza el análisis para determinar que Ansible fue la herramienta adecuada para la automatización y que GNS3 es el simulador adecuado para esta implementación.

En el cuarto capítulo, se presenta los resultados tanto del diseño como de la simulación de la red, en donde se muestra la programación utiliza para la configuración de puertos, protocolos de enrutamiento, políticas de seguridad. Finalmente, el capítulo quinto, se muestra un análisis de los resultados de red diseñada y la verificación de que esta red será eficiente a comparación de realizar una configuración de forma manual.

Palabras clave:

Ansible, Cisco Firewall ASA, automatización, ISP.

ABSTRACT

The purpose of this degree work was the design and simulation of a local network connected to Internet service providers that contains security policies implemented in the ASA Firewall device, and with programming for automated configuration using the open source platform Ansible. . . This network allows automatic configuration of interfaces, MPLS and OSPF routing protocols, and security policies.

Chapter one refers to the introduction of the thesis, which includes the statement of the problem, justification and the objectives for the production of this document. In the second chapter, there is a review of the literature and the theoretical framework, necessary for the implementation of the local network connected to the ISP provider that contains security policies, configured automatically.

In chapter three, the research methodology is found, which specifies the design and research modality that was implemented in the degree work, in addition, the analysis is carried out to determine that Ansible was the appropriate tool for automation and that GNS3 is the appropriate simulator for this implementation.

In the fourth chapter, the results of both the design and the simulation of the network are presented, where the programming used for the configuration of ports, routing protocols, and security policies is shown. Finally, the fifth chapter shows a analysis of the results of the designed network and the verification that this network will be efficient compared to performing a configuration manually.

Keywords:

Ansible, Cisco Firewall ASA, automation, ISP.

CAPÍTULO 1
INTRODUCCIÓN

A lo largo del tiempo, las empresas de telecomunicaciones que brindan servicios de conectividad a Internet han ido evolucionando. una de estas mejoras es la implementación de automatización de la configuración de los equipos y de las técnicas para un mejor procesamiento en la red.

Las redes automatizadas tienen grandes ventajas como la velocidad para implementar cambios dentro de la misma además de recuperar de manera rápida datos requeridos de los dispositivos dentro de la red, permitiendo resolver problemas de forma dinámica. Al realizar la automatización de la red ayuda a la empresa a ser más ágil, además permite que los equipos y operaciones puedan realizarse más rápido que las contrapartes de TI para implementar aplicaciones (Conde, 2021).

Por otra parte, la seguridad de una red es un factor a tomarse en cuenta porque pueden existir diferentes vulnerabilidades desde cualquier computador que esté conectado, es decir que el administrador de la red debe tener en consideración los diferentes ataques o virus que pueden afectar a la red. Por tal razón, se debe verificar las consideraciones, herramientas y políticas de seguridad que permitan proteger la red.

1.1. Planteamiento del Problema

Actualmente, existen varias dificultades que se debe tomar en cuenta al momento del diseño de una infraestructura, como es la seguridad en los equipos, por otra parte, también se debe tener en cuenta la recuperación de manera rápida de datos requeridos de los dispositivos dentro de la red.

En vista que las configuraciones de las redes empresariales son repetitivas y suelen ejecutarse manualmente, presentan demora en implementarse en cada uno de los equipos de la red. Debido a que existen ataques a las redes intentando robar información, es importante la protección de esta red.

Al tener una perspectiva manual tanto para la configuración, como en la actualización de red, suelen ser lentas y propensas a errores, lo que dificulta la necesidad de poder cambiar rápidamente los requisitos de carga de trabajo (Red Hat Inc, 2023).

El Internet es una herramienta indispensable, independientemente de la utilidad que este tiene. Por otra parte, existen usuarios mal intencionados que tratan de causar daño a los consumidores u organizaciones tratando de destruir, secuestrar o robar información preciada.

1.2. Justificación

El poder utilizar herramientas de seguridad en una red ayudará mejorar la defensa de información y datos de diferentes empresas, además al utilizar la automatización de configuración permitirá que los servicios de red sean más ágiles y flexibles.

1.3. Objetivo

1.3.1. Objetivo General

Diseñar y simular una red local conectada a proveedores de servicios de Internet (ISP) con políticas de seguridad y automatización de sus configuraciones.

1.3.2. Objetivos Específicos

- Diseñar la automatización de configuraciones en la red para optimizar y garantizar la disponibilidad.

- Analizar la mejor herramienta de seguridad para implementar en el diseño de la red.
- Evaluar el correcto funcionamiento de la simulación del diseño de la red de las configuraciones automatizadas y la seguridad.
- Establecer indicadores de desempeño que verifiquen la seguridad y la autonomía de la red diseñada.

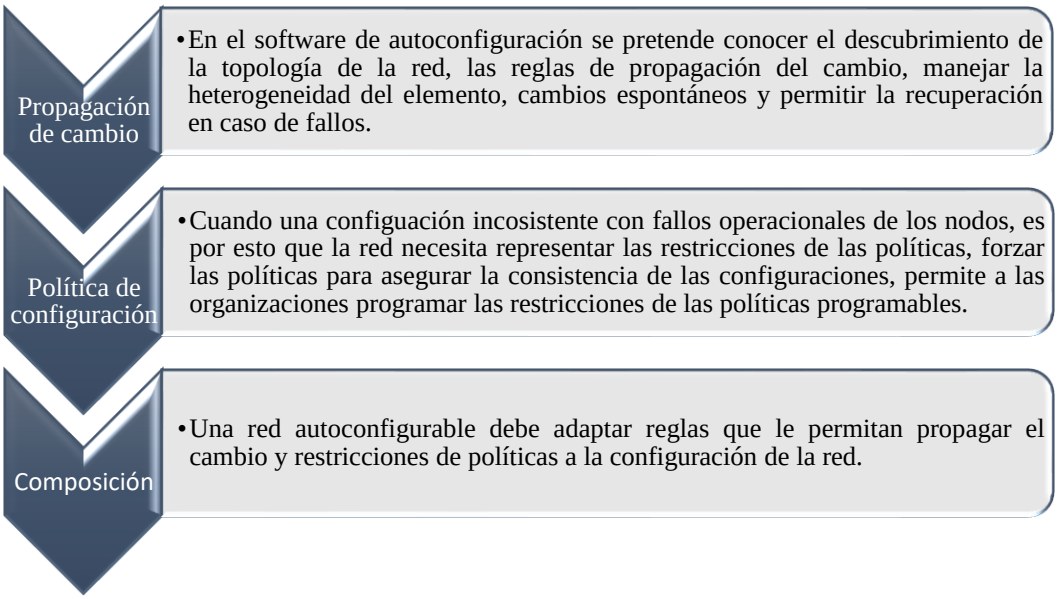
CAPÍTULO 2
REVISIÓN DE LA LITERATURA

Para la automatización de una red se puede utilizar diferentes herramientas de configuración de la administración utilizadas hoy en día, como son: Ansible, Chef, Puppet, Salt Stack, entre otros, los cuales consienten en reunir la administración y configuraciones de los equipos que se encuentran en la red (Yunga, 2018).

2.1. Automatización de la configuración de la red

Figura 1

Factores del diseño de red auto configurable.



Fuente. Tomado de (Yunga, 2018).

La automatización de la configuración es un método que permite configurar, regular, aprovisionar, administrar, y la seguridad de los dispositivos a fin de progresar en la eficacia y eficiencia de una red. Las diferentes ventajas de la automatización son el reducir la complejidad en los procesos, mejoramiento del rendimiento del sistema, crecer la productividad de una empresa de TI, entre otros. Hay que considerar los

diferentes factores a prestar atención para el diseño de una red autoconfigurable, como se muestra en la Figura 1 (Yunga, 2018).

2.2. Herramientas de automatización de la red

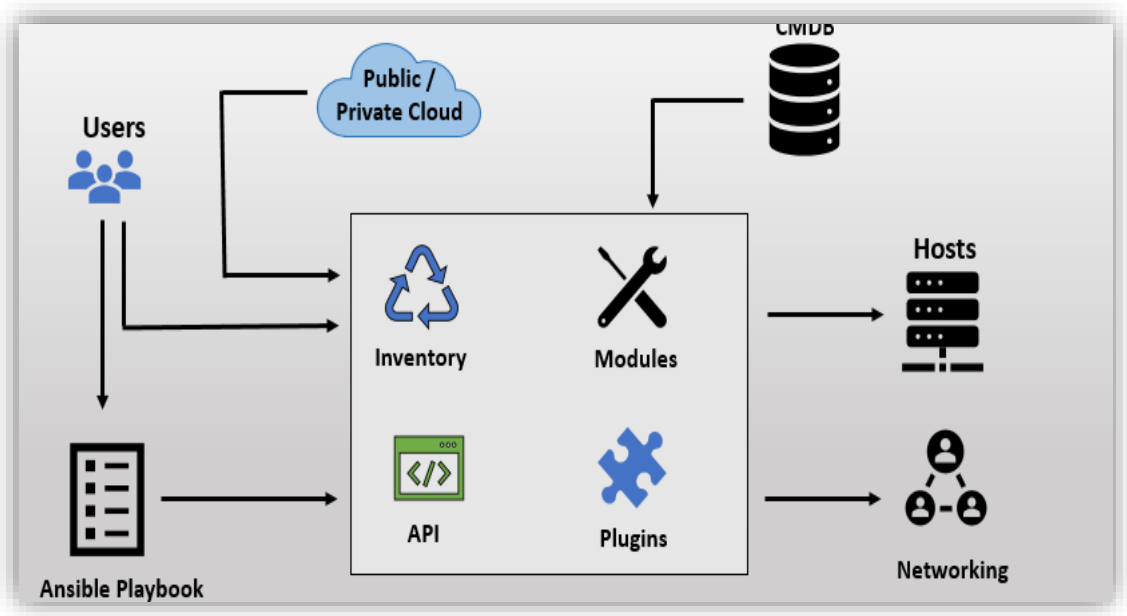
Existen diferentes herramientas, plataformas o protocolos para la automatización de la red entre las que se destacan: Chef, Pupple, Ansible, Salt Talk, entre otras. A continuación, se muestra las características de cada una las aplicaciones mencionadas.

2.2.1. Plataforma de Automatización Ansible

La aplicación Ansible es un programa utilizado para la automatización de TI mediante comando de código abierto programado en Python, encargado de realizar la configuración de sistemas, implementación de software y organización de flujos de trabajo avanzado. Las ventajas de la aplicación es que son de fácil uso, tiene un enfoque en seguridad y confiablidad.

Figura 2

Arquitectura de Ansible



Fuente. Tomado de **(EDUCBA, 2023)**.

La automatización puede ayudar en el aumentar la seguridad y la escalabilidad con configuraciones coherentes, adaptarse a las demandas cambiantes de los clientes, ejecutar actualizaciones, parches y el mantenimiento requerido automáticamente, simplificación en la gestión de la red, reducción las fallas en la red y mejorar sus resultados (Red Hat, 2023).

La arquitectura de Ansible mostrada en la Figura 2, contiene el nodo de control y los hosts administrados. Los hosts a administrase se enumeran en un inventario de host, mientras que el archivo de texto en el nodo de control se tiene los nombres de host administrado. Además, se debe iniciar una sesión y ejecutan Ansible mediante un playbook y un host de destino para administrar.

Inventario. El Ansible en un inventario permite enumerar los host o dispositivos que forman parte de la red. Estos hosts pueden ser servidores, routers, switches, máquinas virtuales, etc. El inventario alcanzará a cada dispositivo mediante conexión ssh, por tal razón se debe tener configurado esta conexión en todos los equipos.

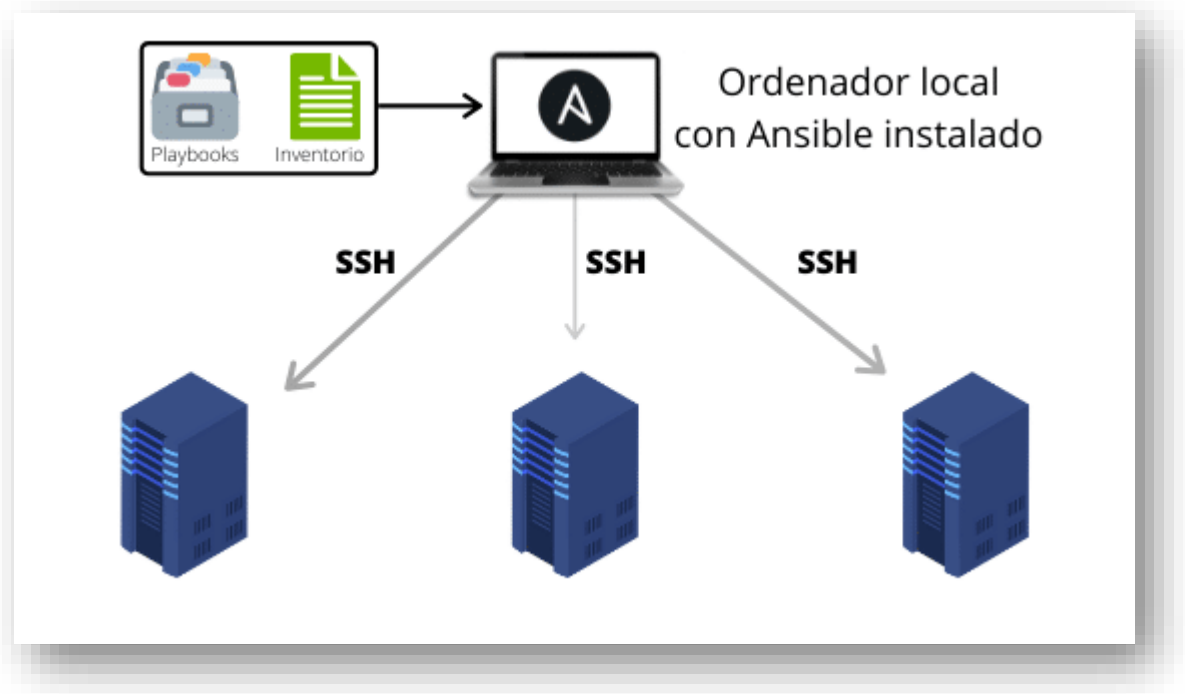
Playbooks. Es un conjunto ordenado de tareas de automatización en formato YAML. Estos archivos, tienen varias tareas, configuraciones y roles que se aplicarán en la máquina que administrará (Red Hat Inc, 2023). Al iniciar Ansible, permite que los administradores verifiquen el estado de la red, y de esta forma conseguir el estado deseado o realizar los cambios necesarios a implementarse en el sistema (Enciso & Morales, 2021).

2.2.2. Servidor de control Ansible

El implementar Ansible en un entorno, se necesita tener servidores o máquinas en las que se ejecutará Ansible y desde las cuales de administrarán la red. Un servidor o máquina virtual que despliegue el control de Ansible. En este server se deberá instalar Ansible y de esta forma poder ejecutar los playbook que ayudarán a gestionar la red.

Figura 3

Servidor de control Ansible



Fuente. Tomado de **(Diéguez, 2020)**

Se puede ejecutar cualquier sistema operativo compatible con Ansible, como Linux (CentOS, Ubuntu, RHEL,entre otros) o incluso macOS. Además, se debe tener Python instalado en este servidor, ya que Ansible está escrito en Python.

Ansible puede gestionar sistemas que ejecuten una amplia variedad de sistemas operativos, incluyendo Linux, Windows, macOS y otros. Además, es importante asegurarse de que el servidor de control tenga acceso SSH a los nodos gestionados.

Configurar un servidor con Ansible es un proceso que implica la automatización de tareas de aprovisionamiento, configuración y gestión de servidores y aplicaciones. Ansible es una herramienta de automatización que utiliza lenguaje YAML para definir las tareas a realizar en un servidor objetivo (Red Hat, 2023).

2.2.3. Chef

Chef se trata de una infraestructura de código abierto que se utiliza para realizar la creación de automatización de políticas, que se tiene en la Figura 4. Al utilizar esta herramienta para automatizar la gestión de configuración, ayuda a definir políticas que sean repetibles, coherentes y reutilizables, permitiendo que exista mayor agilidad y seguridad en una red. Los componentes requeridos para Chef son los siguientes:

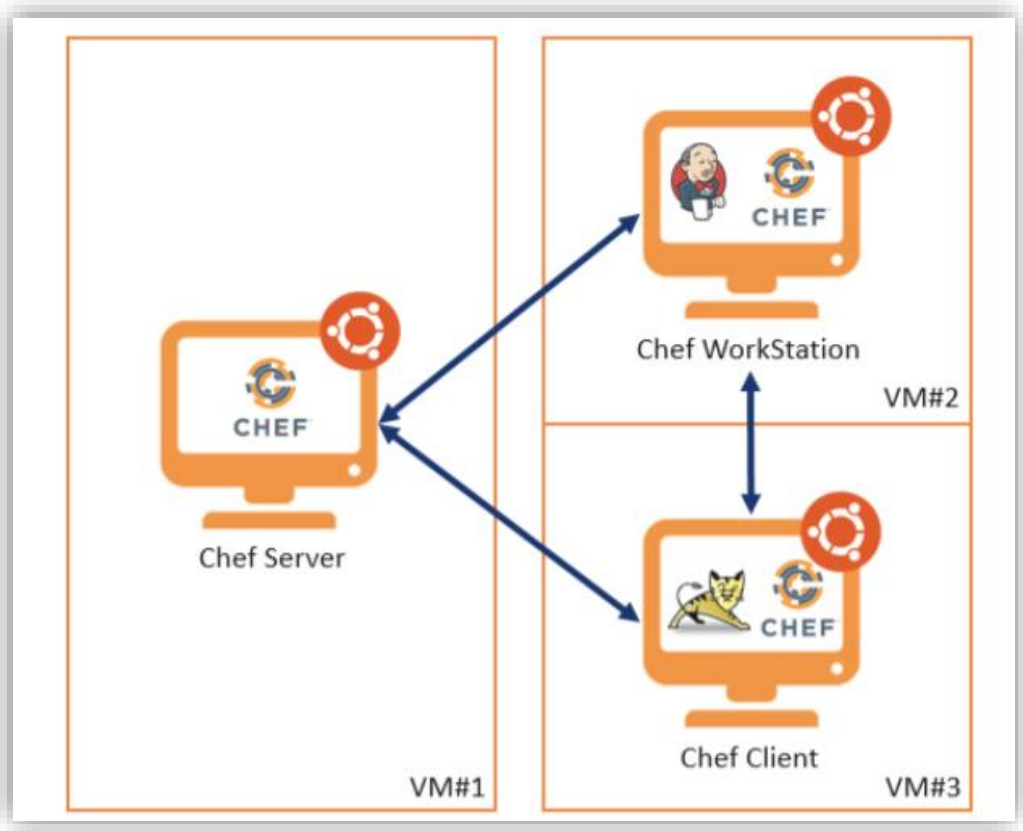
- Chef Workstation: Es donde los desarrolladores y administradores crean y prueban recetas y cookbooks. Incluye herramientas de línea de comandos y una estación de trabajo local.
- Chef Server: El servidor central donde se almacenan los cookbooks y las políticas de configuración. Los nodos se registran en el Chef Server y obtienen las configuraciones y actualizaciones desde aquí.
- Nodos: Son las máquinas o servidores que se administran con Chef. Cada nodo tiene un cliente de Chef instalado que se comunica con el Chef Server para obtener las configuraciones y aplicarlas localmente.
- Cookbooks: Son las unidades básicas de configuración en Chef. Un cookbook contiene recetas que describen cómo configurar y gestionar un componente o servicio específico en un nodo. Las recetas son escritas en un lenguaje específico de Chef.

- Recetas: Son instrucciones escritas en un lenguaje específico de Chef que definen cómo configurar un componente o servicio en un nodo. Las recetas se agrupan en cookbooks.

Para utilizar Chef se tiene tres diferentes componentes Chef Worktation permite crear y probar políticas, Chef Infra logra realizar el cumplimiento del estado del sistema y Chef Automate agrega y valida los datos del sistema (Progress Software Corporation, 2023).

Figura 4

Arquitectura de Chef



Fuente. Tomado de **(Rajesh, 2022)**

Para la creación de políticas de automatización se utiliza chef infra cookbook que se requiere su configuración. También se utiliza libros de recetas para describir los

recursos del sistema que se administran, tales como archivos, plantillas y paquetes del software (Progress Software Corporation, 2023).

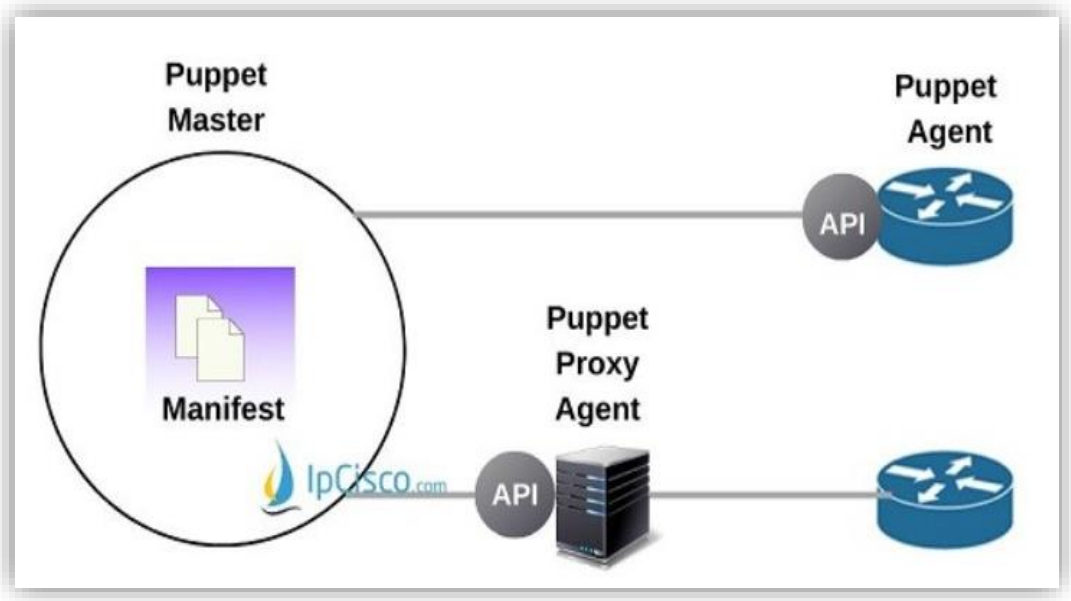
2.2.4. Puppet

Puppet es una infraestructura que permite la administración de configuración automatizada, se caracteriza porque permite mantener la infraestructura como sea definida por el administrador. Esta herramienta permite el ahorro de tiempo a los administradores, como se observa en la Figura 5.

Puppet Bolt ayuda con la ejecución de un script que permite la automatización, además admite el reciclaje del código existente como YAML, PowerShell, Bash, Python, entre otros.

Figura 5

Arquitectura de Puppet.



Fuente. Tomado de **(IPCISCO, 2022)**

A diferencia de Ansible, Puppet utiliza una arquitectura sin agente, es decir no hay ningún agente en el dispositivo remoto, por lo tanto, se utiliza un agente proxy

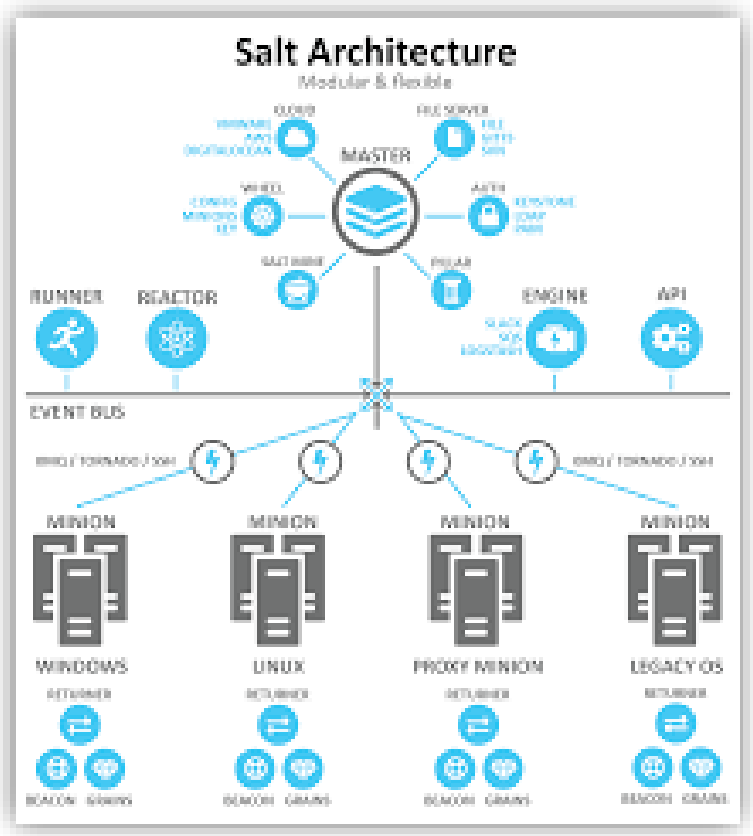
ejecutado en un dispositivo externo. Entonces sus componentes son: Puppet Master, Puppet Agent, catálogos, manifiestos, clase y nodo administrado (Puppet, 2023).

2.2.5. Salt Stack

Salt Stack permite realizar la automatización de redes de la configuración, administración y operaciones de una red informática. Utiliza un repositorio central para suministrar nuevos servidores y otra infraestructura de TI. Su arquitectura está basada en la ejecución de comandos de manera remota, como se observa el ejemplo de la Figura 6.

Figura 6.

Arquitectura de Salt Stack



Fuente. Tomado de **(VMware, 2023)**

Una de las principales características es conocida por la administración paralela, su automatización de datos es en tiempo real, gestión del sistema, en caso de que se

requiera realizar remotamente, se ejecuta en segundo, no en horas o minutos (Yunga, 2018).

Tiene diferente características y conceptos de Saltstack entre las principales son: SaltStack se basa en un modelo de arquitectura maestro-minion, donde el servidor maestro controla y gestiona los nodos minions en la red. El maestro envía comandos y configuraciones a los minions para que los ejecuten.

Utiliza módulos para realizar tareas específicas en los minions. Los estados se utilizan para definir y mantener la configuración deseada en los minions. Los grains proporcionan información sobre los minions, como su sistema operativo o hardware, que puede utilizarse en las configuraciones.

2.3. Políticas de seguridad en red

Estas políticas de seguridad en una red es un conjunto de directrices y procedimientos que se establece proteger la disponibilidad, integridad y confidencialidad de los datos de una organización. Éstas están diseñadas para asegurar que la información se maneje y utilice de manera adecuada y segura, para prevenir la pérdida o divulgación no autorizada de datos.

Algunos elementos comunes, pueden incluirse en estas políticas para desarrollarse en una red:

- Definición de roles y responsabilidades: Para esta política debe establecer las responsabilidades de las personas dentro de la organización con respecto a la relación de la seguridad de la red.

- Control de acceso: La política debe definir cómo se manejará el acceso a la información, quiénes tienen permiso para acceder a ella y cómo se controlará ese acceso.

- Protección de datos: La política debe establecer cómo se protegerá la información en términos de confidencialidad, integridad y disponibilidad.

- Gestión de contraseña: Se debe precisar los requerimientos para la gestión de contraseñas, incluyendo la complejidad, el cambio periódico y la gestión de contraseñas olvidadas.

- Copias de seguridad y recuperación de datos: La política debe establecer los procedimientos de recuperación de información en caso de desastres o incidentes que puedan afectar a la red.

- Uso de dispositivos móviles y medios de almacenamiento: La política debe definir cómo se manejarán los dispositivos móviles y los medios de almacenamiento, como las unidades USB, para evitar la pérdida de datos.

- Política de uso aceptable: La política debe establecer lo que se considera un uso aceptable de los recursos informáticos de la organización, incluyendo el estar en Internet, correo electrónico y el uso de dispositivos móviles.

- Capacitación y concientización: La política debe establecer los requisitos de capacitación para el personal de la empresa pueda comprometerse en los procedimientos y políticas.

- Evaluación y revisión: La política debe definir cómo se llevarán a cabo las evaluaciones y revisiones periódicas de la política para asegurarse de que sigue siendo relevante y efectiva.

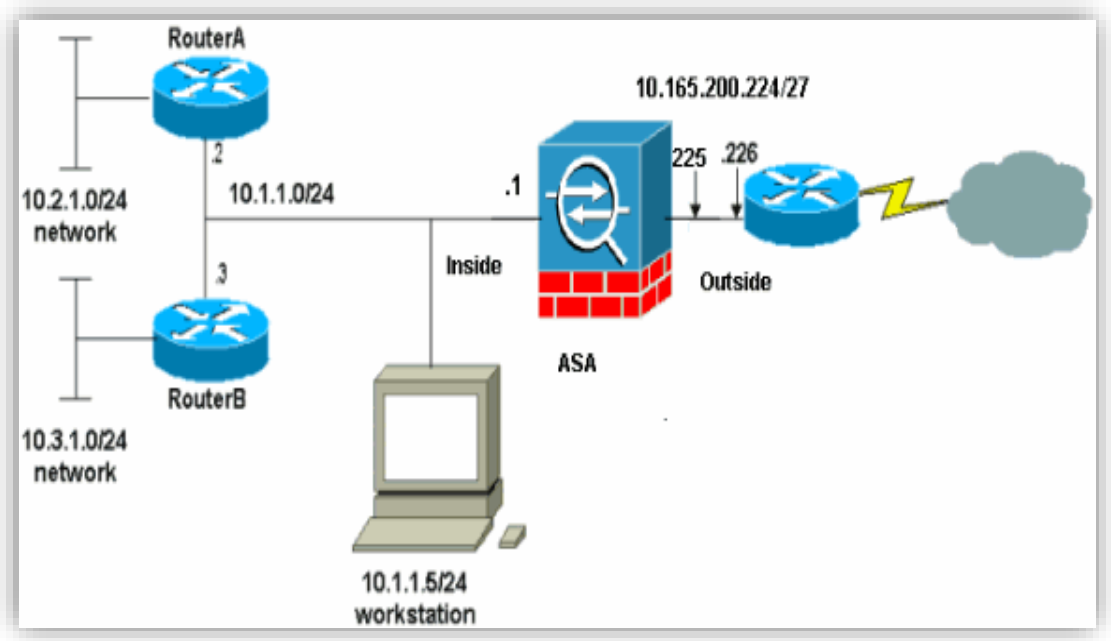
- Estos son solo algunos de los elementos que pueden incluirse en las políticas. Es importante que cada organización adapte su política a sus necesidades específicas y la revise regularmente para asegurarse de que sigue siendo efectiva y relevante (Figueroa, Rodriguez, Bone, & Saltos, 2017).

2.4. Firewall ASA Cisco

Es un dispositivo de seguridad de una red que permite la protección de la mismas de amenazas externas, como por ejemplo ataques de malware, virus, hackers, entre otros. El dispositivo mediante capas de seguridad garantiza el proteger el sistema, incluyendo la inspección de paquetes, la detección de intrusiones y la filtración de contenido.

Figura 7

Firewall Cisco ASA



Fuente. Tomado de **(Cisco, 2023)**

Para proteger sus recursos, debe ver los usuarios, las aplicaciones, los dispositivos y las amenazas presentes en su red y lo que hacen. Los firewalls Cisco

ASA 5500-X proporcionan la visibilidad de red que usted necesita además de una protección superior frente a amenazas y malware avanzado, y mayor automatización para reducir costes y complejidad.

Entre las características del Firewall ASA de Cisco se incluyen:

- Inspección de paquetes: examina la entra y salida de los paquetes de información en la red para verificar posibles amenazas.
- Detección de intrusiones: puede detectar intentos de acceso no autorizado a la red y notificar al administrador del sistema.
- Filtración de contenido: puede bloquear el acceso a ciertos sitios web o tipos de contenido para mantener la privacidad y seguridad de la red.
- VPN: permite la entrada remota segura a la red a través de una conexión VPN (Virtual Private Network).
- Integración con Active Directory: puede autenticar a los usuarios utilizando el sistema de autenticación de Active Directory.
- Configuración basada en roles: permite la configuración basada en roles para limitar el acceso tanto a la administración como a la configuración de la red solo a usuarios autorizados.
- Escalabilidad: es escalable y puede utilizarse para proteger desde redes pequeñas hasta redes grandes.

En general, el Firewall ASA de Cisco es una solución de seguridad de red eficaz y confiable que proporciona múltiples capas de protección para una red. Sin embargo, es importante tener en cuenta que la implementación y configuración adecuadas son

clave para garantizar que el ASA funcione correctamente y proteja la red de manera efectiva (Cisco, 2023).

2.5. Protocolo de enrutamiento en una red.

Los protocolos permiten que los equipos de una red de networking intercambien información sobre redes cercanas para determinar la mejor ruta y de esta forma enviar información entre dispositivos (Muñoz-Gallego, 2018). Esto se efectúa a través de una tabla de enrutamiento, las rutas que se encuentran en la tabla determinarán la mejor ruta para alcanzar la dirección deseada, además de utilizar méticas que ayudar a tomar la decisión de cuál es la mejor ruta, basándose en factores de distancia, velocidad, entre otros. Existen varios protocolos de enrutamiento como son OSPF, RIP, EIGRP, BGP, IS-IS (Ángulo & Camacho, 2006).

2.5.1. Protocolo OSPF

Es uno de los protocolos implementado en las redes mediante el cual se intercambia información detallada sobre la topología y de esta forma, calcular la opción más óptima para él envió de paquetes. Además, este protocolo es un IGP generalmente utilizado dentro de un sistema autónomo en redes empresariales (IBM Corporation, 2021).

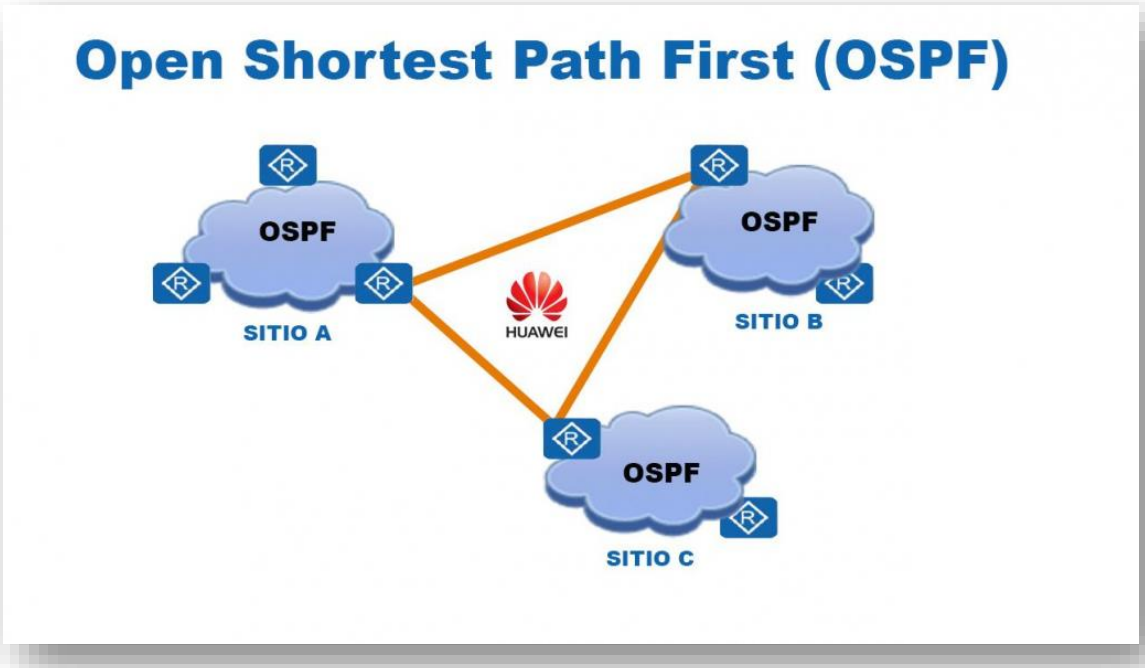
Una de las ventajas a considerar es que sirve en redes de gran tamaño, permitiendo recalcular la mejor ruta en un tiempo corto cuando cambia la topología, a comparación de otras protocolos (IBM Corporation, 2021).

Es altamente escalable y adecuado en redes de distintos tamaños. Permite dividir redes grandes en áreas de forma lógica. Es decir que cada área tendrá su propia tabla de

enrutamiento, que está conectada a través de una red backbone que será el área 0, mejorando de esta forma eficiencia y escalabilidad e OSPF (Valverde, 2016).

Figura 8

Protocolo de enrutamiento OSPF



Fuente. Tomado de **(Bolaños, 2023)**

A continuación, se muestran las diferentes características de OSPF:

Cálculo de Rutas: Para determinar las rutas más cortas en la red, utiliza el algoritmo SPF (Shortest Path First), garantizando que la ruta seleccionada sea óptima y eficiente (IBM Corporation, 2021).

Áreas OSPF: OSPF divide una red grande en áreas lógicas. Cada área tiene su propia tabla de enrutamiento y está conectada a través de un área de backbone (área 0). Este enfoque mejora la eficiencia y la escalabilidad de OSPF.

Métrica: OSPF utiliza una métrica llamada "costo" para determinar la mejor ruta. El costo se basa en la velocidad de la interfaz y se utiliza para calcular la ruta más eficiente.

Convergencia Rápida: mediante este protocolo, al existir un cambio en la topología, permite que se adapte a cambios eligiendo la mejor ruta, y de esta forma actualizando la tabla de enrutamiento (Flórez, 2019).

Autenticación: este protocolo permite garantizar la seguridad en la comunicación entre routers de la red, lo cual se logra mediante clave de autenticación (Flórez, 2019).

Tipos de Mensajes OSPF: el protocolo utiliza diferentes tipos de mensajes para la transmisión de enrutamiento, entre los cuales se tiene: paquetes Hello, descripción de base de datos “DBD”, solicitud de estado de enlace “LSR”, actualización de estado de enlace “LSU” y reconocimiento de estado de enlace “LSAck” (Gil Martinez, 2015).

2.5.2. Protocolo EIGRP

El protocolo se utiliza en redes empresariales, para el intercambio de paquetes maneja un protocolo de transporte fiable, mediante el algoritmo equilibrado e hibrido, además utiliza una métrica compuesta. Fue desarrollado por Cisco Systems, como una versión renovada de IGRP (Mier & Mier, 2008). Existen diferentes características de EIGP:

Métrica avanzada: su métrica se basa en confiabilidad, ancho de banda, retardo y carga para calcular las mejores rutas. Esto permite tomar decisiones de enrutamiento más precisas.

Actualizaciones parciales: envía actualizaciones parciales en lugar de enviar la tabla de enrutamiento completa en cada momento, lo que permite la reducción del ancho de banda.

Distribución de vecinos: Los routers vecinos intercambian información de enrutamiento, lo que les permite mantener una topología de red actualizada y responder rápidamente a los cambios en la red.

Soporte para IPv4 e IPv6: puede ser utilizado tanto en redes IPv4 como en IPv6, lo que lo hace versátil para entornos que migran a la nueva versión del protocolo IP.

Detección de fallos rápida: es capaz de detectar fallos en las rutas y cambiar a rutas alternativas de manera rápida, lo que contribuye a una recuperación más rápida en caso de problemas de red.

Autenticación: ofrece opciones de autenticación para garantizar que las actualizaciones de enrutamiento sean seguras y provengan de fuentes confiables.

Compatibilidad con VLSM y CIDR: es compatible con la división de SRes de longitud variable (VLSM) y la agregación de rutas mediante la notación de máscara de longitud de prefijo.

Balanceo de carga: permite distribuir el tráfico de la red entre múltiples rutas iguales de manera equitativa o de igual costo para mejorar la utilización de la red (Mier & Mier, 2008).

2.6. MPLS (Multiprotocol Label Switching)

El método MPLS se utiliza para el enrutamiento de paquetes en una red que mejora la eficiencia y la velocidad al momento de realizar la comunicación de datos en redes grandes (Guevara, 2010). Es considerado un protocolo de conmutación de

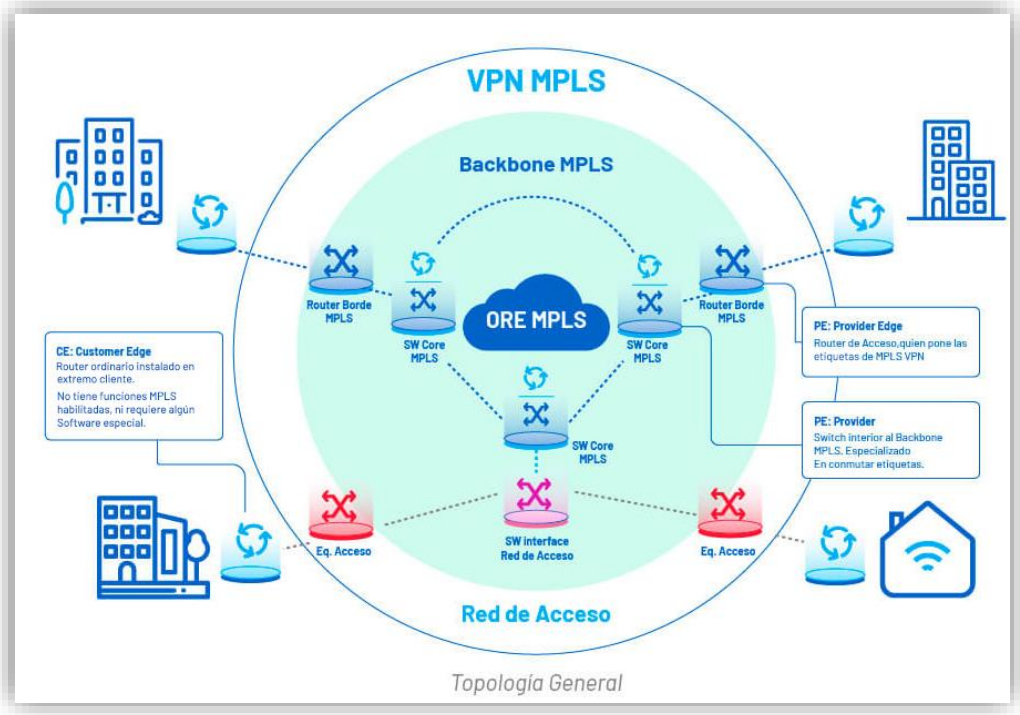
etiquetas utilizado para enruta y conmutar paquetes de forma más ágil que un enrutamiento IP. Además, se tiene las siguientes características principales de esta tecnología (De Ghein, 2016).

La conmutación de etiquetas se basa en que cada paquete de datos se coloca con una etiqueta corta antes de ingresar a la red MPLS. Estas etiquetas se utilizan para definir la ruta que el paquete debe seguir a través de la red, lo que mejora el desempeño de la red, aumenta la calidad de servicio, entre otros (Guevara, 2010).

MPLS se utiliza principalmente en la capa de red (modelo OSI) y ofrece varias ventajas en términos de enrutamiento, calidad de servicio y administración de tráfico.

Figura 9

Ejemplo red MPLS



Fuente. Tomado de **(Entel, 2023)**

El enrutamiento basado en etiquetas a diferencia del enrutamiento IP tradicional, en el que los routers toman decisiones de enrutar mediante la dirección IP de destino.

Además, toma decisiones de enrutamiento basadas en las etiquetas asociadas a los paquetes. Esto permite un enrutamiento más rápido y menos complejo.

Los túneles LSP (Label-Switched Paths), permite la creación de estos túneles a través de la red. Los cuales pueden ser utilizados para enrutar tráfico de manera eficiente y establecer caminos dedicados para aplicaciones específicas, como QoS o la administración de tráfico.

Calidad de Servicio: admite la ejecución de políticas de calidad de servicio que permiten priorizar el tráfico de la red. Siendo esto de utilidad para garantizar el rendimiento necesario para aplicaciones sensitivas a la latencia, como, por ejemplo, videoconferencia o VoIP.

Por otra parte, MPLS es escalable y se adapta bien a redes grandes. Los proveedores de servicios de Internet utilizan este protocolo para administrar el tráfico de manera eficiente en sus redes globales.

También, accede a la segmentación de red en dominios de etiquetas (VRFs, Virtual Routing and Forwarding). Esto es útil para crear redes virtuales independientes en una infraestructura de red compartida.

La Seguridad es compatible con técnicas, como VPN (Virtual Private Network), las cuales permiten avalan la seguridad de datos en tránsito y su privacidad.

2.7. Simulador de redes de computadoras.

Los simuladores son herramientas de software en donde se pueden crear entornos de red de forma virtual, simulando un entorno real. Estos simuladores permiten visualizar de forma gráfica los equipos que se requerirían en una red. Son utilizados

para fines académicos, investigativos, de planificación y para la resolución de algún problema.

En la actualidad, en el mercado existen diferentes programas, que se tiene en la Figura 10, son la simulación de redes y la elección de la herramienta depende de las necesidades, capacidad y características que se presentan a continuación (Pacheco, 2022):

Figura 10

Simuladores para redes de computadora

Fuente. Tomado de **(Agapidis, 2023)**

Packet Tracer: esta herramienta es utiliza en el proceso de aprendizaje en redes de computadoras, ciberseguirdad, e internet de las cosas, desarrollado por la empresa Cisco. Generalmente utilizados para la certificación Cisco CCNNA.

GNS3: Esta plataforma de código abierto permite la creación de topologías y ejecución de sistemas operativos de routers y switches reales en máquinas virtuales, utilizado para la emulación de redes complejas.

EVE-NG: Es una tecnología que permite emular dispositivos de diferentes fabricantes, trabaja en la nube y realiza virtualizaciones dinámicas que permite familiarizarse con ambientes de trabajo reales.

OMNeT++: Es un framework de simulación de código abierto que ayudaen la modelación de redes, protocolos y aplicaciones. Es personalizable y generalmente utilizado en proyectos académicos y científicos.

OPNET: Esta herramienta ayuda en el analices y comparación de redes de diferentes tecnologías, soporta varios protocolos como TCP, IPv6, VoIP, entre otros, Opnet es parte de Riverbed Modeler.

Boson NetSim: Es un simulador que permitir utilizar equipos de manera virtual, tiene como característica que puede utilizar nodos en la nube. Proporciona variedad de escenarios de red para practicar y aprender.

Para la elección más adecuada herramienta de simulación es recomendable verifica cada una de las características y capacidades que tienen estas plataformas y de esta forma comparar cual será la mejor elección que se adapte a los objetivos propuestos.

2.8. Indicadores de desempeño KPI

Los indicadores de clave KPI (Key Performance Indicators), son métricas que son realizadas con el fin de valorar el rendimiento y el éxito de una empresa, proyecto, proceso o actividad en función a sus metas. Los KPI son fundamentales para medir el progreso hacia los resultados deseados y tomar decisiones informadas.

Estos indicadores son datos numéricos o métricas específicas que se pueden medir y cuantificar. Estas medidas se relacionan con distintos aspectos del desempeño,

como puede ser la calidad del producto, rendimiento financiero, satisfacción del cliente, eficiencia operativa, entre otros.

Se miden de manera continua o en intervalos definidos a lo largo del tiempo. Esto permite realizar un seguimiento del desempeño en el transcurso de un proyecto, ciclo de negocio o período determinado (Rios, 2019).

En el contexto de redes de computadoras y telecomunicaciones, los Indicadores Clave de Desempeño (KPIs) son métricas críticas que ayudan a evaluar y supervisar el rendimiento de la red. Estos KPIs proporcionan información esencial para garantizar un funcionamiento óptimo de la red y una experiencia satisfactoria para los usuarios.

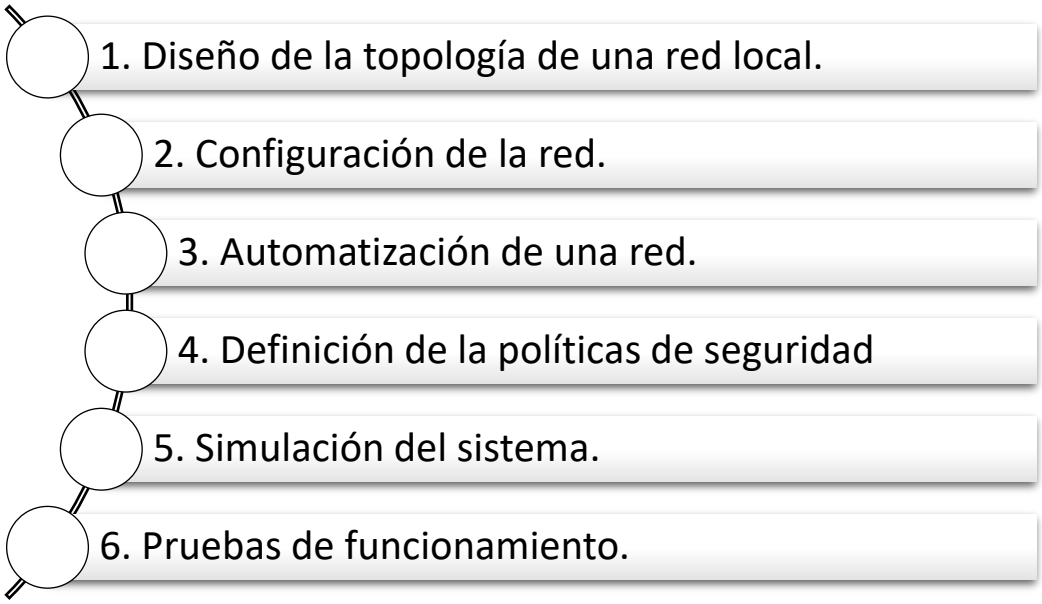
CAPÍTULO 3
METODOLOGÍA

3.1. Marco Metodológico

A continuación, se va a desarrollar el diseño de una red local, que tenga políticas de seguridad implementadas en el dispositivo Firewall Cisco ASA, además de utilizar la herramienta en un servidor Ansible para la automatización de la red y verificar el mejoramiento de la misma. Como se muestra en la Figura 11, los distintos pasos a implementarse en la metodología del diseño de la red.

Figura 11.

Metodología a utilizarse en el desarrollo de la red.



En primer lugar, para el cumplimiento del primer objetivo se realizará el diseño de la topología. Se prevé efectuar una red con conexiones a ISP con protocolos de enrutamiento MPLS y OSPF. Una vez diseñada la red, se verifica la configuración a usarse en los equipos, para posteriormente implementar la automatización del sistema mediante Ansible.

Para la ejecución del segundo objetivo, se recolectará información referente a las diferentes herramientas de seguridad y las políticas de seguridad que se van a implementar en el diseño del sistema. En este caso se utilizará un Firewall ASA Cisco, que permita proteger la red de amenazas externas, implementando las políticas de seguridad mediante ACL (Access Control List), de esta forma accediendo o negando el tráfico de red según una variedad de criterios y reglas de NAT (Network Address Translation) para traducir las direcciones y puertos de origen y destino en el tráfico de red.

Para llevar a cabo el tercer objetivo, se verifica la simulación del diseño de la red en el software de simulación GNS3 el cual permita validar el correcto funcionamiento de las configuraciones automatizadas y la verificación de la seguridad implementada en los dispositivos.

Como objetivo final, se realizará la implementación de indicadores de rendimiento de la red que permitan garantizar la disponibilidad de la misma y el correcto funcionamiento de la automatización y las políticas de seguridades ya implementadas.

3.2. Herramienta de automatización.

En este momento, existe diferentes herramientas que permite la automatización de una red en el mercado, por tal razón se realizará una comparación entre tres de las mismas “Ansible”, “Puppet” y “SaltStack”, para que se pueda determinar la más acorde al diseño del sistema que se implementará. Este diseño se verificará las diferentes especificaciones, enfoques y características que tiene cada una de las herramientas.

Tabla 1

Comparación de las herramientas de automatización de la red.

CARACTERÍSTICAS	ANSIBLE	PUPPET	SALT STACK
Sistema operativo	Linux, Windows	Linux, Mac OS, Windows	Linux (requiere licencia)
Licencia	GNU	Apache	Apache
Lenguaje de programación	Python	Ruby	Python
Código fuente	Abierto	Abierto	Abierto
Agente/ sin agente	Sin Agente	Sin Agente	Sin Agente (Agentes Proxy)
Protocolo de comunicación	SSH	SSH/ WinR	SSH y SCP
Lenguaje de escritura	YAML	YAML/ Lenguaje Puppet	YAML/ SLS (Salt State File)
Arquitectura	Cliente/ Servidor	Cliente/ Servidor	Cliente/ Servidor
Escalabilidad	Altamente escalable	Altamente escalable	Altamente escalable
Soporte en la nube	Todo	Todo	Todo

Fuente. Tomado de **(Wågbrant & Dahlén, 2022)**, **(Yunga, 2018)**, **(Brikman, 2016)**

Tabla 2

Características a tomar en cuenta para la automatización de la red.

	ANSIBLE	PUPPET	SALT STACK
Complejidad de la red	Redes pequeña y sencilla	Redes complejas y distribuidas	Redes con capacidad de gestión granular y escalabilidad
Experiencia del equipo	Si el equipo ya tiene experiencia con alguna de estas herramientas		
Integración con otros sistemas	Considerar la integración de automatización con otros sistemas y herramientas utilizadas en tu red.		
Soporte y comunidad	La herramienta en mención, tiene un gran ayuda en soporte, proporcionando la resolución de problemas y la ayuda en el futuro.		

Nota. Tomado de **(Wågbrant & Dahlén, 2022)**, **(Yunga, 2018)**, **(Brikman, 2016)**

Para la selección de la herramienta adecuada para esta red, se recomienda tener los aspectos mencionados en la Tabla 2. Dado estas características, al implementarse la

red se decide el utilizar para la automatización el software Ansible el cual es utilizado para un red pequeña y sencilla, y cuenta con varia información en el Internet.

Ansible es el apropiado a utilizarlo en esta red debido a su capacidad para automatizar tareas, el gestionar la configuración, mejorar la seguridad y facilitar la administración de los equipos de seguridad críticos de las redes.

3.3. Herramienta de emulación.

Para realizar la simulación del diseño de la red, depende de varios factores como la complejidad del diseño, los requisitos de simulación, la familiaridad con la herramienta y el costo. En la Tabla 3, se mencionará diferentes características que permitan elegir el emulador adecuado.

Tabla 3

Comparación de las herramientas de emulación.

CARACTERÍSTICAS	CISCO PACKET TRACER	GNS3	EVE – NG
Complejidad de la red	Diseño para educación y la simulación de redes simples y medianamente complejas.	Adecuado para simulaciones avanzadas y redes complejas	Adecuado para simulaciones avanzadas y redes complejas
Dispositivos y características admitidas	Se centra principalmente con dispositivos Cisco	Más flexible en compatibilidad con diferentes dispositivos	Más flexible en compatibilidad con diferentes dispositivos
Recursos del sistema	Necesita menos recursos de hardware	Requiere más recursos en redes grandes y complejas	Requiere más recursos en redes grandes y complejas
Licencia y costos	Es gratuito para uso educativo	Ofrecen versiones gratuitas con limitaciones	Ofrecen versiones gratuitas con limitaciones
Interfaz y facilidad de uso	Interfaz gráfica de fácil uso	Requiere conocimiento técnico, ofrece mayor flexibilidad y control	Requiere conocimiento técnico, ofrece mayor flexibilidad y control

Fuente. Tomado de **(Pacheco, 2022)**, **(Agapidis, 2023)**

En cuanto a la opción a elegir la herramienta de simulación, se debe tener en el computador en donde se va a ejecutar debe contar con los suficientes recursos y

potencia. En este caso, se tomó a consideración el tipo de procesador de al menos 4 núcleos, un mínimo de 8GB de RAM, espacio de memoria suficiente para almacenar los playbooks, disco SSD y Windows 11.

Como se observa en la Tabla 3, los simuladores más adecuados a utilizar son GNS3 y EVE-NG para este diseño, sin embargo, se toma la decisión de utilizar GNS3, porque ofrece compatibilidad con máquinas virtuales como VMware, que se necesita para el uso de Ansible. Además, los recursos que utiliza son más bajos en comparación de EVE-NG.

3.4. Selección de las políticas de seguridad a implementar.

Para la selección de las políticas de seguridad en el diseño de una red, se debe considerar diferentes factores para garantizar la protección adecuada de los activos y los datos del sistema. En este caso se tendrá en cuenta las siguientes políticas que se tiene en la Figura 12.

Figura 12

Políticas de seguridad a implementarse en el diseño de la red.

Como primer punto se debe identificar los activos más críticos de la red, como datos confidenciales, servidores, dispositivos de red y sistemas clave. Adicional, se debe evaluar los posibles peligros a los que están exponen estos activos como ataques, de malware, acceso no autorizado, robo de datos, etc.

También, en la red se definirán los objetivos específicos que se lograrán con las políticas de seguridad, entre ellos se tendría a personalizar el acceso a recurso humano dependiendo de los recursos. Proteger datos confidenciales contra filtraciones y garantizar la integridad de las comunicaciones.

Para realizar las políticas de seguridad, hay que tener en cuenta que deben ser claras y específicas, teniendo las metas establecidas. Las políticas deben ser comprensibles y se indicará que acciones de se deberán tomar en caso de alguna violación. Por otra parte, se deberá tener un sistema de monitoreo que permita probar la eficacia de las políticas, además de identificar posibles brechas o áreas de mejora. Finalmente, se debe realizar pruebas y simulaciones de incidentes de seguridad para evaluar cómo actúa la red y verificar la efectividad de las políticas.

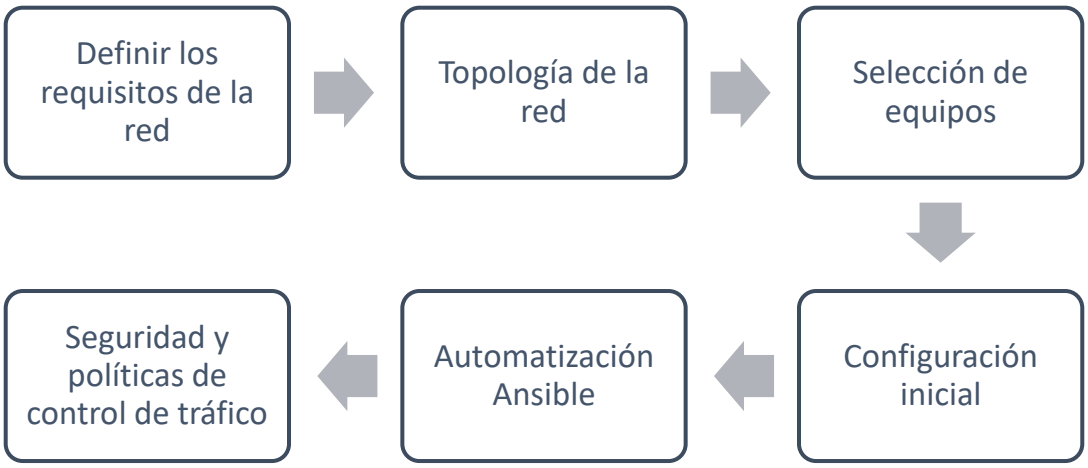
3.5. Diseño de la red a implementarse.

Para el diseño de esta red se tomó en consideración que los equipos sean compatibles, las medidas de seguridad y los dispositivos sean los adecuados para la implementación de los protocolos, como se observa en la Figura 13.

Es un proceso complejo que requerirá planificación, comprensión de todos los requerimientos y desafíos específicos. Para el comienzo del diseño de red se debe definir los requisitos y objetivos, los niveles de seguridad.

Figura 13

Diseño de la red.



Como siguiente punto será el diseño de la sistema de la red, incluyendo la infraestructura física y lógica, segmentación de la red, como se automatizará la red y que políticas de seguridad se podrían ejecutar.

Después, se procede a seleccionar el tipo de equipos adecuados para la simulación, como son el routers, switch, firewalls y sistemas de gestión que cumplan con las necesidades de la red y permitan la compatibilidad con Ansible. El definir las políticas de seguridad serán la base para la protección de la red y de la información. Esto puede incluir la detección y prevención de intrusiones, monitorización y la gestión de vulnerabilidades.

Una vez obtenido la red a implementarse, en los equipos se procederá a realizar la configuración inicial, en este caso será la configuración de todos los enlaces con su direccionamiento IP para permitir la conexión hacia Ansible.

El siguiente paso, será configurar Ansible como herramienta de automatización para la gestión de la red y de esta forma aplicar las políticas de seguridad. Es decir que realiza la configuración de los inventarios y playbooks necesario para que funcione correctamente la red.

En cuanto se tenga se tenga el correcto funcionamiento de la red automatizada de procederá a realizar el desarrollo de las políticas de seguridad que aborden aspectos como la autenticación, el cifrado, segmentación y prevención de ataques.

Finalmente, se realizarán las pruebas de funcionamiento para corroborar que la red implementada cumple con los requisitos y objetivos planteados inicialmente en el diseño de la red.

3.6. Indicadores

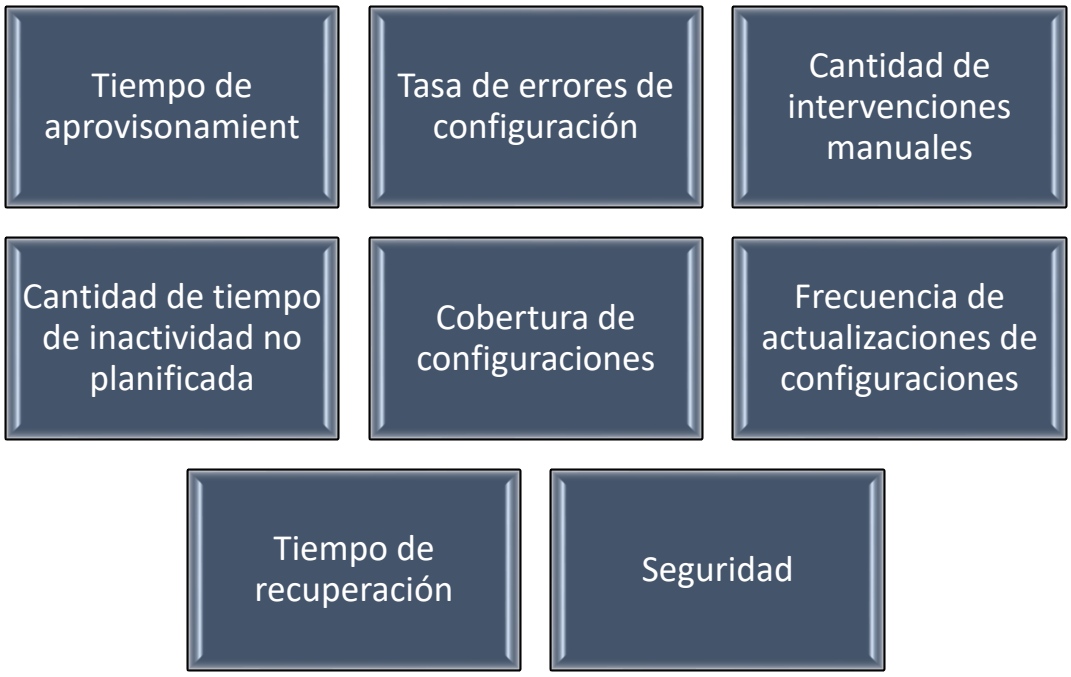
Para una red gestionada con Ansible pueden variar según los objetivos específicos de la organización y el nivel de automatización implementado. Sin embargo, aquí hay algunos KPIs comunes que se pueden utilizar para evaluar el rendimiento y la eficacia de la gestión de la red con Ansible.

Tiempo de aprovisionamiento: Mide el tiempo necesario para implementar nuevos dispositivos de red o para aplicar cambios en la configuración de la red. Un tiempo de aprovisionamiento más corto indica una implementación más rápida y ágil.

Tasa de errores de configuración: Cuantifica la cantidad de errores de configuración encontrados en la red después de que se haya aplicado una automatización con Ansible. Un menor número de errores sugiere una mayor precisión y fiabilidad en las tareas automatizadas.

Figura 14

Tipos de KPI comunes para utilizar para la red.



Cantidad de intervenciones manuales: Mide la cantidad de veces que los administradores de red deben intervenir manualmente para corregir problemas o realizar cambios después de que Ansible haya aplicado la configuración. Menos intervenciones manuales indican una mayor eficiencia de la automatización.

Cantidad de tiempo de inactividad no planificado: Registra el tiempo de inactividad no planificado de la red. Ansible ayuda a mantener una configuración consistente y evitar errores humanos, lo que debería reflejarse en un tiempo de inactividad reducido.

Cobertura de configuración: Evalúa la proporción de dispositivos de red en la infraestructura total que están bajo la gestión y automatización de Ansible. Una alta cobertura significa que más dispositivos están siendo administrados de manera eficiente.

Frecuencia de actualización de configuraciones: Mide la frecuencia con la que se actualizan las configuraciones de red mediante Ansible. Un mayor número de actualizaciones puede sugerir una respuesta más ágil a los cambios y mejoras en la red.

Tiempo de recuperación: Mide el tiempo necesario para recuperar la red de fallos o incidentes. Ansible puede ayudar a restaurar la configuración correcta rápidamente y, por lo tanto, reducir el tiempo de recuperación.

Seguridad: Evalúa la eficacia de las políticas de seguridad implementadas a través de Ansible. Puedes medir la cantidad de vulnerabilidades o problemas de seguridad detectados y solucionados por Ansible.

CAPÍTULO 4

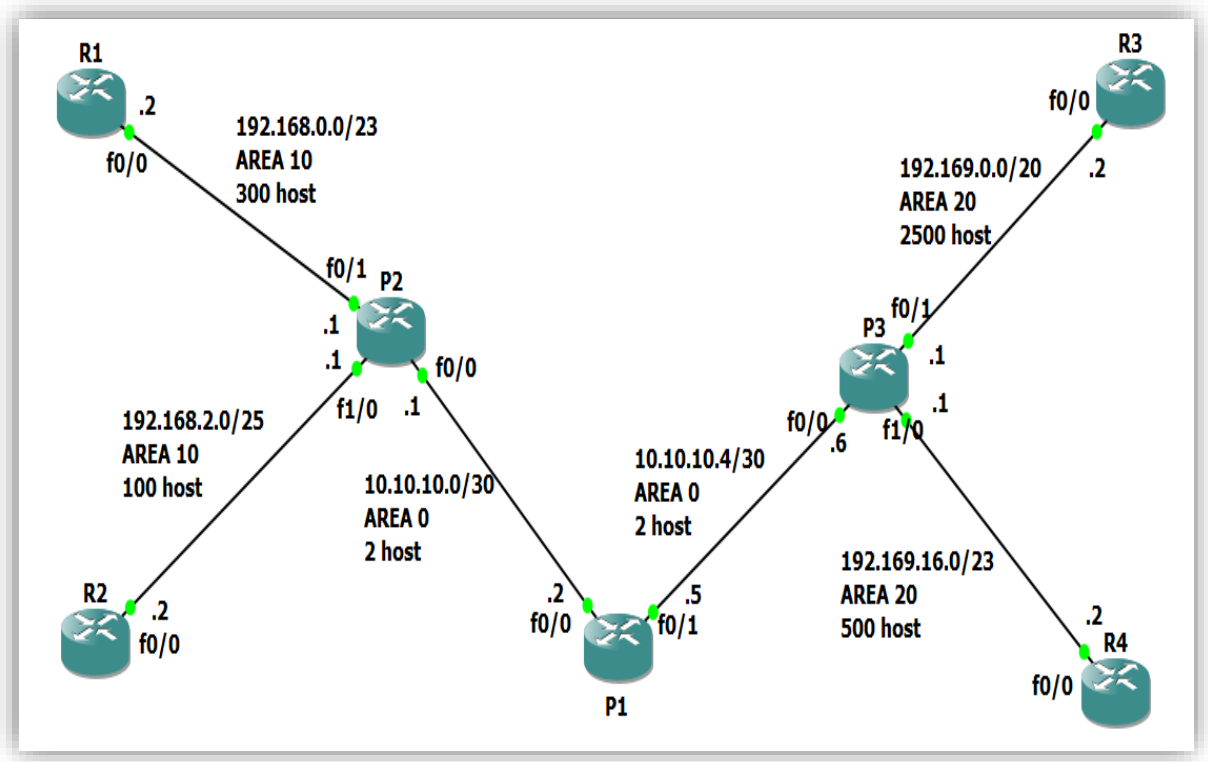
RESULTADOS

4.1. Diseño de la red configuración de forma manual

Para el diseño de la red es importante mencionar que realizará la configuración MPLS y OSPF, que se muestra en la Figura 15, para lo cual se debe tener en cuenta que se realiza el VLSM para el direccionamiento IP que se muestra en la Tabla 4, en donde se asume una cantidad de host para los diferentes equipos de cada red.

Figura 15

Topología inicial de la red**.**



4.1.1. Direccionamiento IP.

Los equipos R1 y R2, pertenecen a las 192.168.0.0 en el área 10, se asume la cantidad de host mostrados anteriormente, mientras que los equipos R3 y R4 forman parte de la 192.169.0.0 del área 20, por otra parte, la red 10.10.10.0 corresponden los

equipos P1, P2 y P3, que pertenecerá al área 0. En la Tabla 5, se tiene las direcciones IP que pertenecen a cada puerto.

Tabla 4

Tabla de Enrutamiento del diseño de la red.

SR	# HOST	MÁSCARA	DIRECCIÓN DE RED	BROADCAST	RANGO
SR 1	300	23	192.168.0.0	192.168.1.255	192.168.0.1 – 192.168.1.254
SR 2	100	25	192.168.2.0	192.168.2.127	192.168.2.1 – 192.168.2.126
SR 3	2500	20	192.169.0.0	192.169.15.255	192.169.0.1 – 192.169.15.254
SR 4	500	23	192.169.16.0	192.169.17.255	192.169.16.1 – 192.169.17.254
SR 5	2	30	10.10.10.0	10.10.10.3	10.10.10.1 – 10.10.10.2
SR 6	2	30	10.10.10.4	10.10.10.7	10.10.10.5 – 10.10.10.6

En la Tabla 5, se muestra el direccionamiento IP de la topología y los requerimientos mencionados anteriormente. Cada interfaz muestra el puerto al que pertenecen en los routers, en donde se tiene 3 SRes diferentes.

Tabla 5

Direccionamiento IP de la red.

Equipo	*Puerto*	*Dirección IP*	*Área*
P1	Fa 0/0	10.10.10.2	0
P1	Fa 0/1	10.10.10.5	0
P1	Fa 1/0	10.10.10.9	0
P2	Fa 0/0	10.10.10.1	0
P2	Fa 0/1	192.168.0.1	10
P2	Fa 1/0	192.168.2.1	10
P3	Fa 0/0	10.10.10.6	0
P3	Fa 0/1	192.169.0.1	20
P3	Fa 1/0	192.169.16.1	20
R1	Fa 0/0	192.168.0.2	10
R2	Fa 0/0	192.168.2.2	10
R3	Fa 0/0	192.169.0.2	20
R4	Fa 0/0	192.169.16.2	20

4.1.2. Elección del protocolo de enrutamiento.

Para la selección del enrutamiento de la red de proveedores ISP, se pueden utilizar varios protocolos, dependiendo de las necesidades y la topología. Entre estos factores se tiene los requisitos de seguridad, escalabilidad, compatibilidad de equipos, y servicios existentes, entre otros.

Tabla 6

Comparación de protocolos de enrutamiento.

Características	EIGRP	OSPF	ISIS
Protocolo	Vector de distancia	Estado de enlace	Estado de enlace
Estándar	Compatibilidad con cisco	Compatibilidad con varios equipos	Compatibilidad con varios equipos
Convergencia	Convergencia rápida en redes pequeñas y medianas	Convergencia rápida	Convergencia rápida en redes grandes
Escalabilidad	En redes pequeñas y medianas	En redes grandes y complejas	Se utiliza en redes grandes
Métrica	Métrica el costo basado en la velocidad del enlace	Métrica compuesta	Métrica fija, un costo fijo

Nota. Tomado de **(Huawei, 2022)**, **(Mier Ruiz & Mier Ruiz, 2008)**, **(Telecapp Inc., 2023)**

Como se observa en la Tabla 6, se tiene una comparación de los distintos protocolos de enrutamiento que se puede utilizar en esta red ISP. OSPF está basado en estándares abiertos que son compatibles con diferentes equipos de distintos fabricantes, esto significa que si tu red ISP utiliza una variedad de dispositivos de diferentes proveedores.

Este protocolo es relativamente simple de configurar y administrar, especialmente en comparación con IS-IS o EIGRP. Además, se tiene una amplia base de conocimientos en este protocolo, una abundancia de herramientas de monitoreo y administración disponibles. A comparación de EIGRP, es compatible con dispositivos de diferentes fabricantes, ofrece una mayor flexibilidad en términos de operatividad,

facilita la implementación de políticas de enrutamiento, por tal razón, se elige el protocolo OSPF (Telecapp Inc., 2023).

Por otra parte, MPLS facilita la segmentación y el etiquetado de tráfico, que se utiliza generalmente para una red de ISP. En esta red y con la automatización mediante Ansible, MPLS puede mejorar la eficiencia operativa, confiabilidad y escalabilidad, al tiempo que simplifica la gestión y la implementación de servicios, que serán de utilidad para varios clientes y servicios con tareas complejas. Además, es conveniente si se requiere un sistema de monitoreo del estado de la red, ya se para diagnóstico y solución de fallos (Díez Álvarez, 2016).

4.1.3. Elección de los equipos

Elegir los equipos adecuados para la simulación de la red permitirá reducir las fallas al implementar a la red con el sistema de automatización Ansible y el firewall ASA para la ejecución de las políticas de seguridad. Se debe tomar en cuenta que los equipos sean compatibles con el simulador GNS3, verificar que los equipos sean compatibles con Ansible, verificar la facilidad de configuración y comprobar que exista suficiente información para el desarrollo de la infraestructura.

Cisco C3725: es un equipo Cisco serie 3700, este router de alto rendimiento con una amplia gama de funciones de enrutamiento y seguridad. El router incluye interfaces Ethernet, serial, WAN por lo que permite su uso en una variedad de escenarios de red. Es compatible con Ansible por lo que permite la comunicación de entre los equipos. Por tal motivo este equipo se ha seleccionado para la implementación de la red.

Network Automation Ansible: Esta herramienta de automatización de código abierto, para automatizar la configuración, gestión y aprovisionamiento de dispositivos y servicios de red. Ansible es comúnmente utilizado en servidores Linux y Windows.

4.1.4. Aplicación de los protocolos de enrutamiento.

En primera instancia, se realiza la configuración de las direcciones IP cada una de las interfaces y el enrutamiento mediante OSPF en cada uno de los equipos y posteriormente con la configuración MPLS en todos los equipos, que se tiene en la Figura 16.

Figura 16

Configuración de OSPF y MPLS en los equipos.

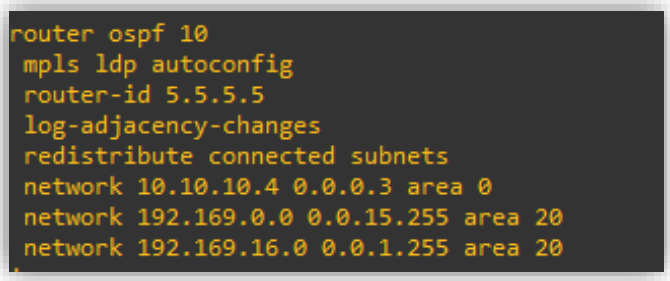
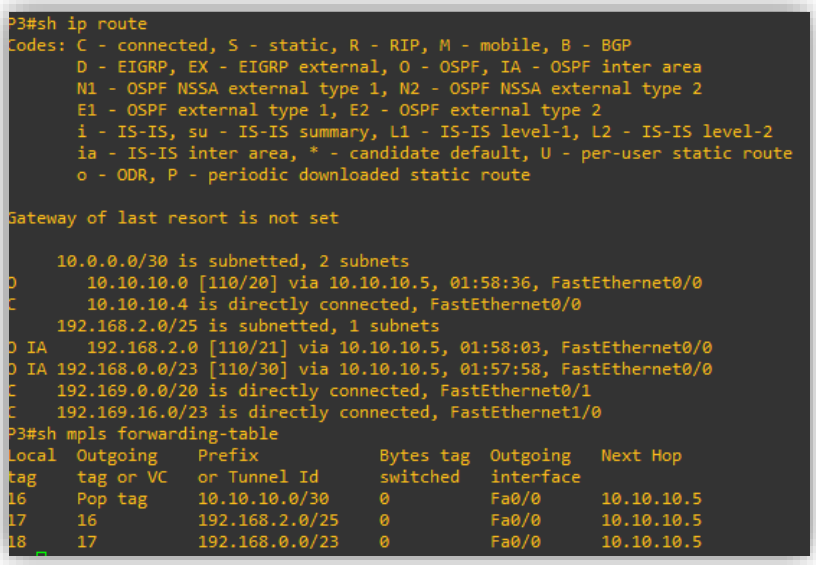


Figura 17

Tabla de enrutamiento y verificación de MPLS



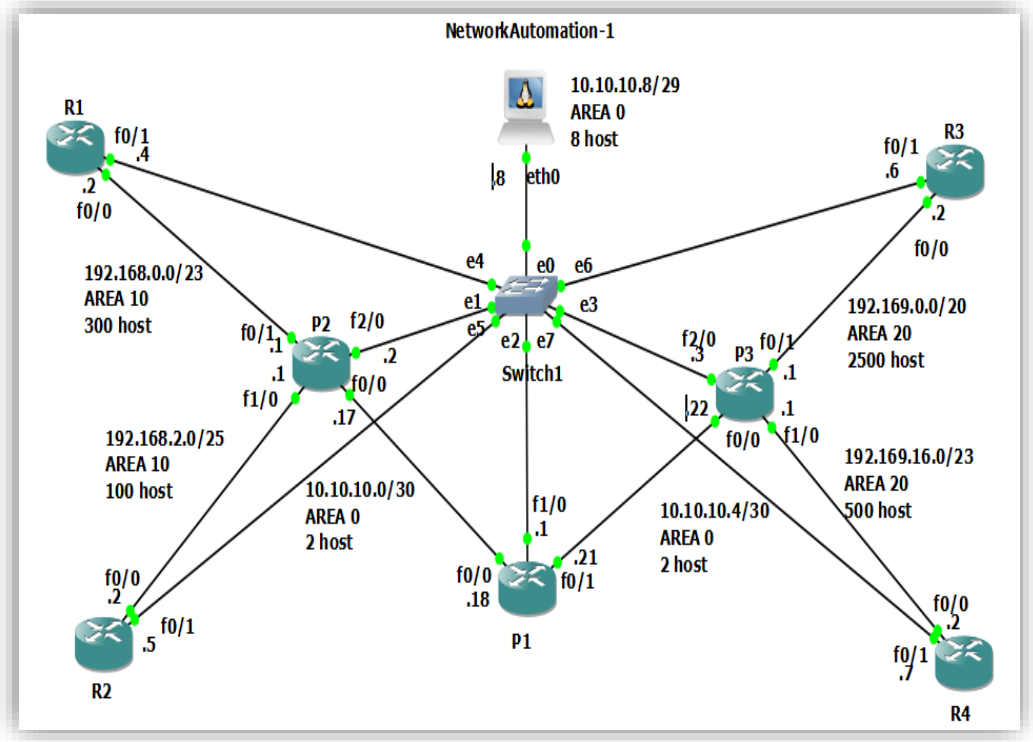
Como se observa en la Figura 17, se obtiene una tabla de enrutamiento mediante la cual se tendrá ping de extremo a extremo en los equipos finales. Además, se podrá observar las etiquetas que utiliza MPLS en la red con el comando *sh mpls forwarding-table*.

4.2. Automatización de la red.

Una vez realizada la configuración OSPF y MPLS, se procederá a implementar la automatización de la red mediante Ansible. En el diagrama de la Figura 18, se agregará este servidor, en el cual se implementa los inventarios necesarios para las configuraciones del sistema de a red.

Figura 18

Diseño de la red con servidor Ansible



En el diseño al momento de incluir Ansible, se incluye una red adicional para la conexión de los equipos al server, como se observa en la Tabla 7 y en la Tabla 8 se tienen los interfaces incluidos de los routers.

Tabla 7

Tabla de Enrutamiento del diseño de la red de forma automática.

SR	# HOST	MÁSCARA	DIRECCIÓN DE RED	BROADCAST	RANGO
SR 1	300	23	192.168.0.0	192.168.1.255	192.168.0.1 – 192.168.1.254
SR 2	100	25	192.168.2.0	192.168.2.127	192.168.2.1 – 192.168.2.126
SR 3	2500	20	192.169.0.0	192.169.15.255	192.169.0.1 – 192.169.15.254
SR 4	500	23	192.169.16.0	192.169.17.255	192.169.16.1 – 192.169.17.254
SR 5	8	28	10.10.10.0	10.10.10.15	10.10.10.1 – 10.10.10.14
SR 6	2	30	10.10.10.16	10.10.10.19	10.10.10.17 – 10.10.10.18
SR 7	2	30	10.10.10.20	10.10.10.23	10.10.10.21 – 10.10.10.22

En este caso se ha agregado una nueva red, en donde se tenga la conexión al servidor Ansible, como se observa en la tabla 7, esta red será la 10.10.10.0/28 con Área 0, en la cual se agregará el equipo mencionado. Para la implementación de la automatización en la red, se debe colocar el direccionamiento IP, y habilitar la conexión ssh que permita el acceso a los routers desde Network Automation Ansible, como se observa en la Figura 19, se tiene el ejemplo de la configuración para el equipo P1.

Figura 19

Configuración de las interfaces en el router P1

```
P1#conf term
Enter configuration commands, one per line.  End with CNTL/Z.
P1(config)#int fa 0/0
P1(config-if)#ip add 10.10.10.18 255.255.255.252
P1(config-if)#no shut
P1(config-if)#exit
P1(config)#int fa 0/1
P1(config-if)#ip add 10.10.10.21 255.255.255.252
P1(config-if)#no shut
P1(config-if)#exit
P1(config)#int fa 1/0
P1(config-if)#ip add 10.10.10.1 255.255.255.240
P1(config-if)#no shut
P1(config-if)#exit
P1(config)#username mayra privilege 15 secret tesis123
P1(config)#line vty 0 4
P1(config-line)#transport input all
P1(config-line)#login local
P1(config-line)#exit
P1(config)#ip domain-name tesis.com
P1(config)#crypto key generate rsa
```

Tabla 8

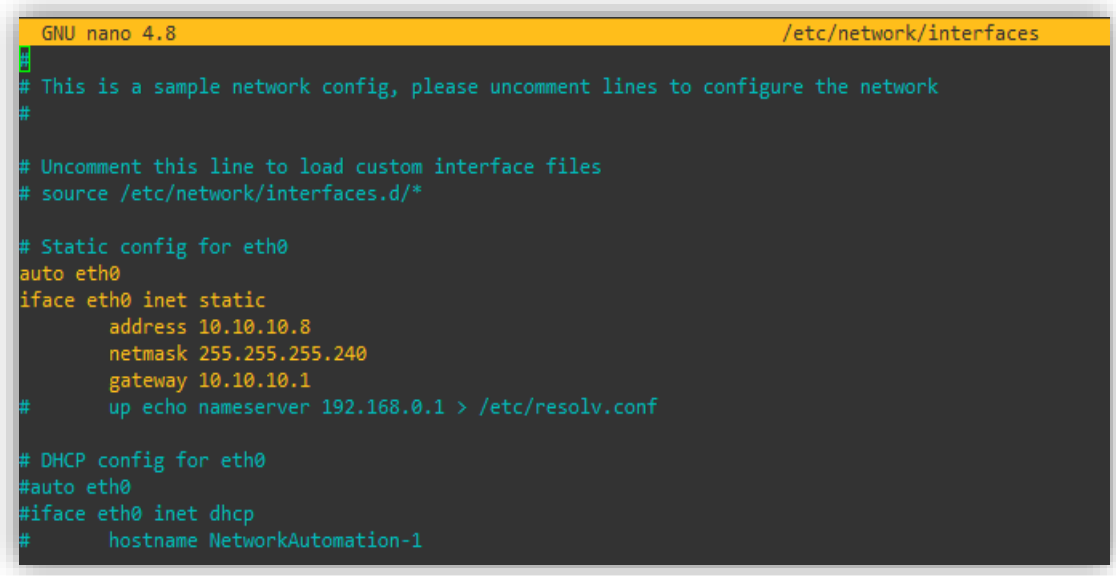
Direccionamiento IP de las interfaces de la red automática.

Equipo	*Puerto*	*Dirección IP*	*Área*
P1	Fa 0/0	10.10.10.18	0
P1	Fa 0/1	10.10.10.21	0
P1	Fa 1/0	10.10.10.1	0
P2	Fa 0/0	10.10.10.17	0
P2	Fa 0/1	192.168.0.1	10
P2	Fa 1/0	192.168.2.1	10
P2	Fa 2/0	10.10.10.2	0
P3	Fa 0/0	10.10.10.22	0
P3	Fa 0/1	192.169.0.1	20
P3	Fa 1/0	192.169.16.1	20
P3	Fa 2/0	10.10.10.3	0
R1	Fa 0/0	192.168.0.2	10
R1	Fa 0/1	10.10.10.4	0
R2	Fa 0/0	192.168.2.2	10
R2	Fa 0/1	10.10.10.5	0
R3	Fa 0/0	192.169.0.2	20
R3	Fa 0/1	10.10.10.6	0
R4	Fa 0/0	192.169.16.2	20
R4	Fa 0/1	10.10.10.7	0

En cuanto se tenga la configuración de las interfaces, se debe asignar la dirección IP de manera manual al server, permitiendo la conectividad con los equipos, en el servidor debemos abrir el archivo con el comando *"nano /etc/network/interfaces"* para asignar la IP como se observa, en la Figura 20, en donde la dirección IP asignada para el servidor Ansible es 10.10.10.8, que se encuentra mediante un switch conectado a todos los equipos de la red que permitan el acceso.

Figura 20

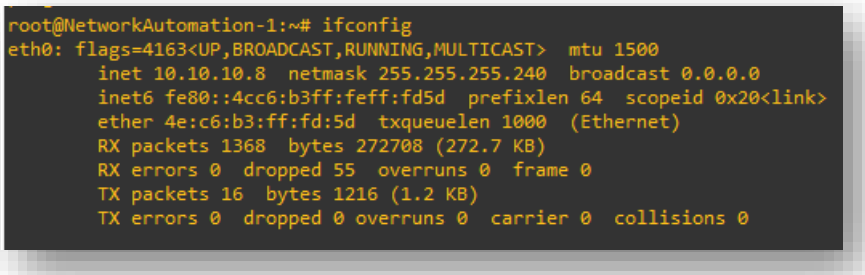
Agregar dirección IP en el servidor Ansible



Hay que tomar en cuenta que para se actualice el direccionamiento en el servidor, se debe reiniciar. Como se observa en la Figura 21, para la comprobación de la dirección IP se debe ejecutar los comandos ***"show ip -a"*** o ***"ifconfig"***, dependiendo del sistema operativo en el que se encuentra Ansible.

Figura 21

Revisión dirección IP estática

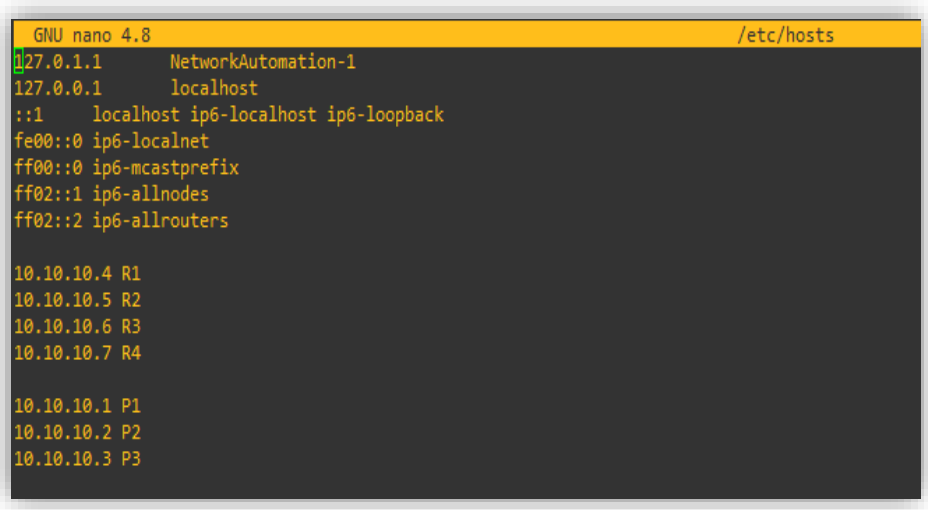


Como se muestra en la Figura 22, al ingresar en el servidor, se debe ejecutar el comando ***"nano /etc/hosts"***, se debe incluir el listado de direcciones IP de los equipos con su respectivo nombre, 10.10.10.1 P1, 10.10.10.2 P2, 10.10.10.3 P3, 10.10.10.4 R1,

10.10.10.5 R2, 10.10.10.6 R3 y 10.10.10.7 R4, es decir todas las direcciones de las interfaces que se encuentran conectadas a Ansible.

Figura 22

Listado de direcciones IP en el servidor Ansible.



Para que pueda existir una conexión segura, es necesario utiliza SSH, por tal razón se requiere que ASA puede establecer una conexión a cada dispositivo y configurar ciertos parámetros.

Posteriormente se debe comprobar si tenemos ping hacia todos los equipos, además del acceso a cada uno de ellos. También, hay que tener en cuenta que para la conexión se realiza mediante el usuario y contraseña configurados en los equipos con anterioridad. Como se observa en la Figura 23, tenemos ping y se logra el ingreso por ssh a los equipos R1 y R4.

Por otra parte, para poder enviar información a los equipos desde el servidor, se debe crear un archivo llamado inventario *"hosts"*, en donde se coloca una etiqueta a los equipos y el nombre de las IP que posteriormente serán llamados en el programa playbook, como se observa en la Figura 24.

Figura 23

Comprobación acceso a diferentes equipos de la red.

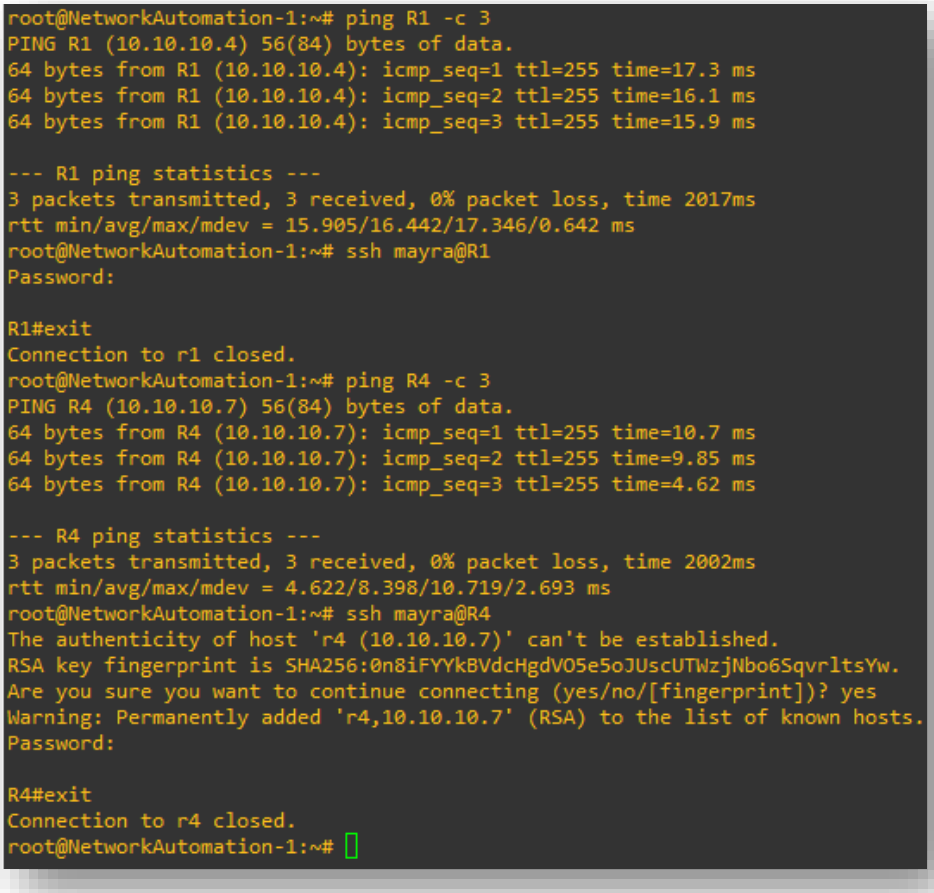
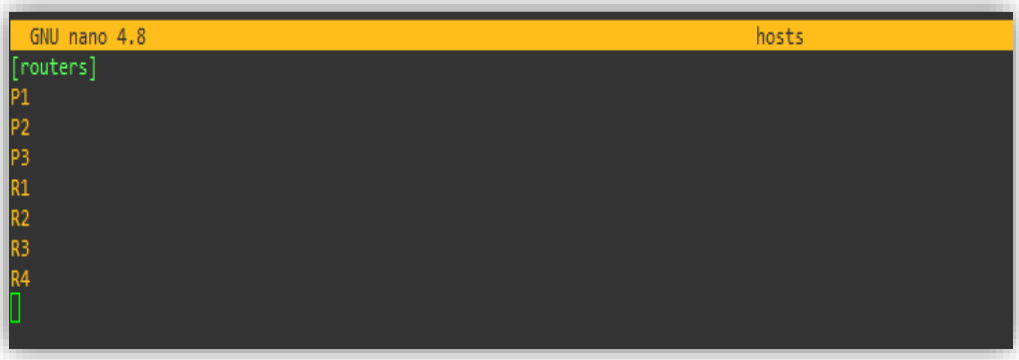


Figura 24

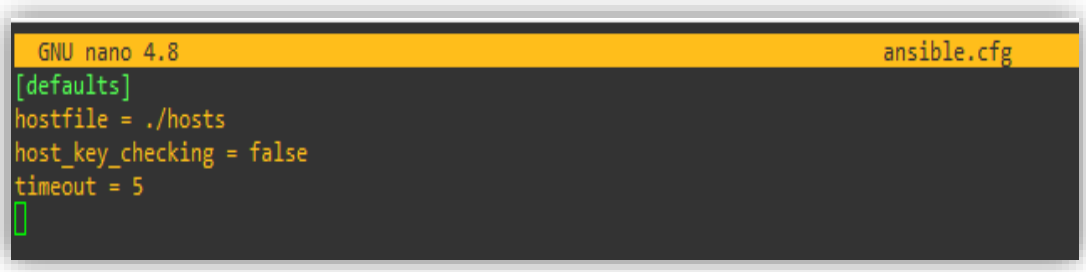
Archivo inventario hosts



Además, se requiere un archivo de configuración ansible en este caso *"ansible.cfg"*, que se muestra en la Figura 25, este contiene los parámetros generales del archivo del inventario y el servidor Ansible.

Figura 25

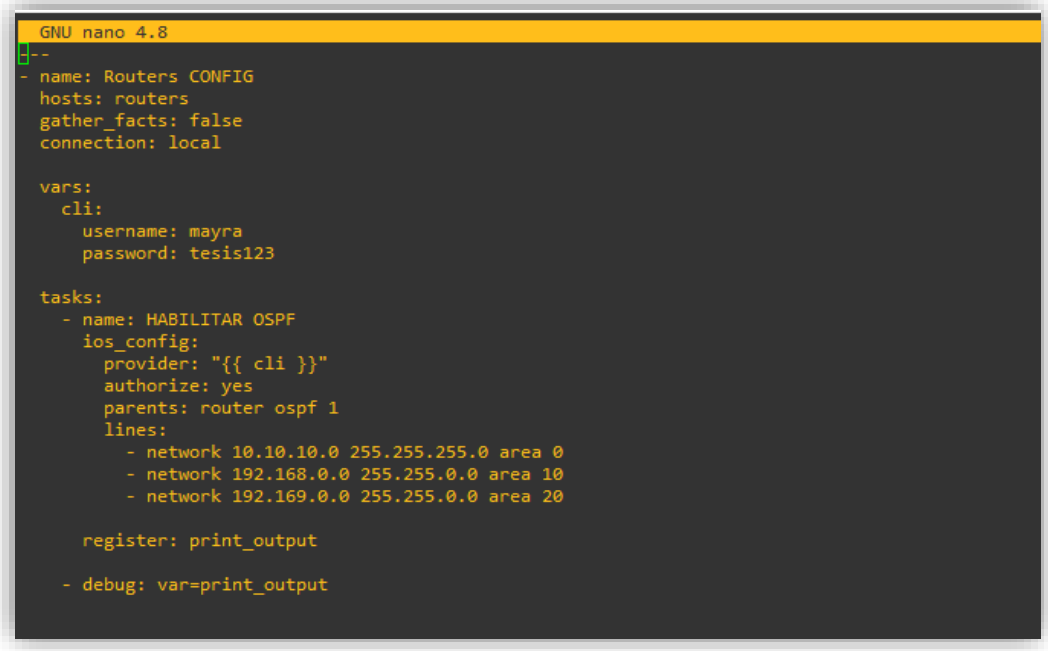
Configuración ansible.cfg



Al tener configurado los archivos antes mencionados, se procede a crear los archivos o archivo playbook con extensión .yml, en este archivo se tiene la configuración del enrutamiento OSPF, de todos los equipos de las red a implementarse con Ansible.

Figura 26

Configuración inicial en playbook.



En el archivo que se observa en la Figura 26, se muestran las líneas de código que se explican a continuación:

- **name:** Describe el nombre de la tarea que se va a realizar

- **hosts:** Especifica el grupo o dispositivos que se ejecutarán en el playbook, lo cuales se encuentran en el archivo del inventario.
- **gather_facts:** Es encargado de configurar si debe recopilar hechos sobre el sistema.
- **connection:** será el tipo de conexión que se tendrá.
- **vars:** se define variables que se utilizarán en el playbook.
- **cli:** permite definir líneas de comando que se ejecutará en el dispositivo.
- **tasks:** define una lista de tareas que se ejecutarán.

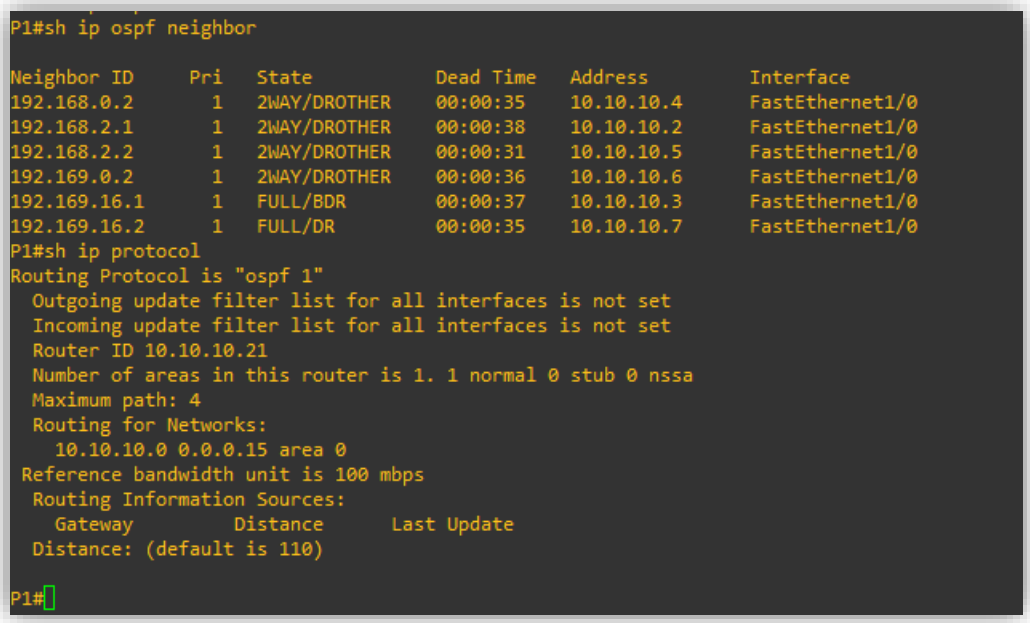
Dentro de una tarea se tiene las siguientes líneas de código:

- **ios_config:** permite enviar comando a la configuración global de los dispositivos IOS.
- **provider:** logra especificar el usuario, contraseña, dirección IP, puertos, entre otros.
- **authorize:** indica si la tarea se debe ejecutar con privilegios de superusuario o elevados.
- **lines:** define los comandos que se enviarán a los dispositivos, en este caso la configuración del ospf.
- **parents:** se especifica en donde se realizará la configuración. En este caso en la ospf 1 para enviar los comandos de las direcciones a aprender.
- **register:** permite guardar los resultados de la tarea.

Una vez implementado los playbook se ejecuta para aplicar las configuraciones requeridas en la red mediante el comando "*ansible-playbook -i hosts configuracion.yml*", esto permite que los equipos ejecuten los comandos necesarios para el enrutamiento OSPF, observado en la Figura 26.

Figura 27

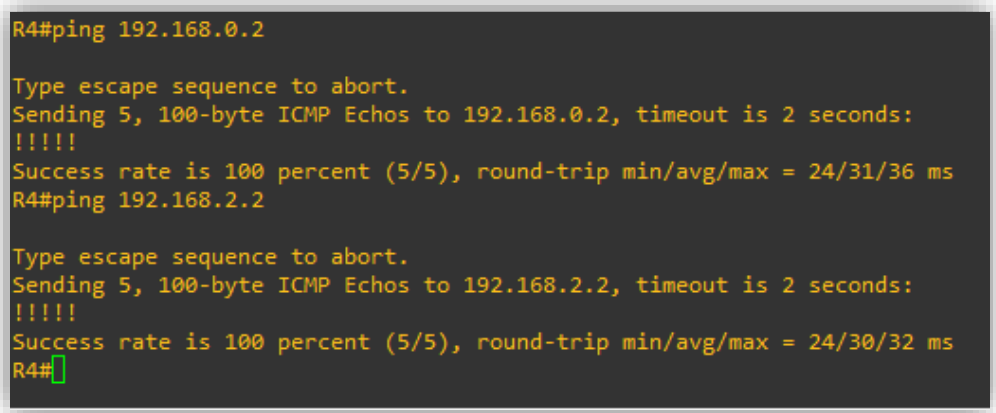
Validación de la configuración en el equipo P1



Para verificar que la configuración este realizada, se debe validar en los dispositivos mediante comandos CLI o utilizando la herramienta de monitoreo. En este caso validamos en el P1 que se muestra en la Figura 27, donde se verifica las rutas aprendidas por OSPF.

Figura 28

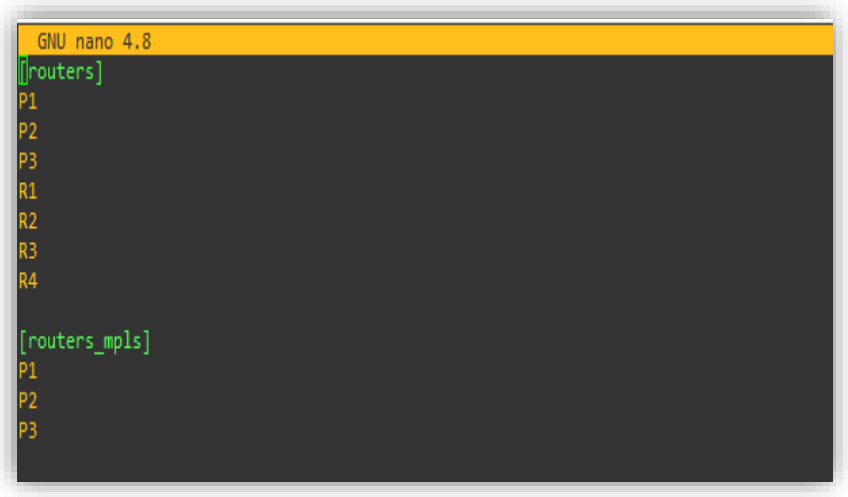
Ping entre los equipos R1 y R2



En la Figura 28, se aprecia la conexión establecida entre los extremos de los equipos de diferentes redes en los routers R1 y R4, por lo tanto, se prueba que se aprendió las rutas necesarias para llegar de un punto a otro.

Figura 29

Etiqueta para el host de MPLS



En cuanto, se ha implementado el protocolo OSPF, se realiza la configuración MPLS con la ayuda de Ansible. Se procede a agregar en el inventario *"host"* los routers que se van a implementar este protocolo, su etiqueta es *[routers_mpls]*, como se tiene en la Figura 29.

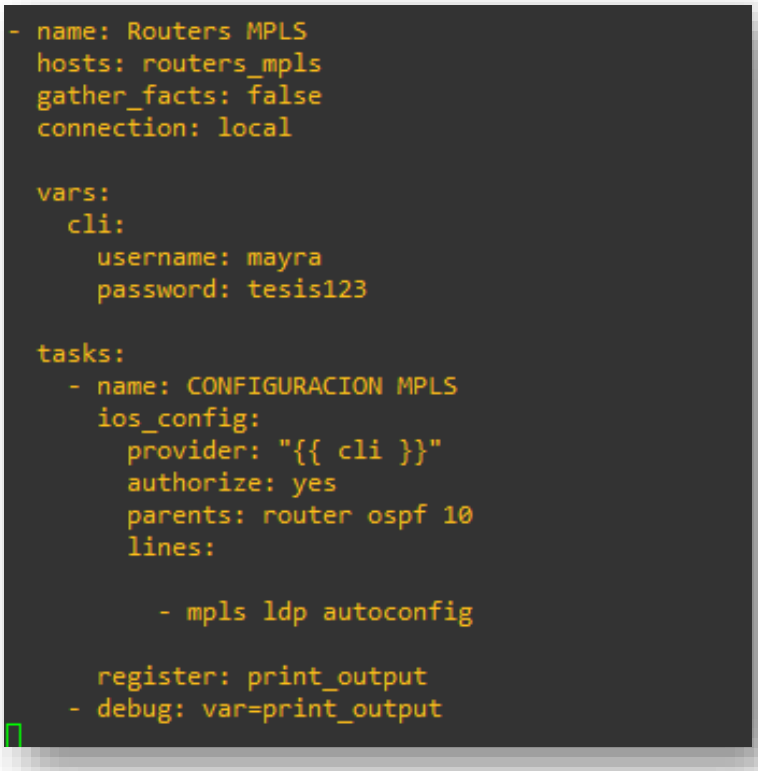
En el playbook, se agrega la configuración MPLS, en este caso se colocará el comando ***"mpls ldp autoconfig"***, como se muestra en la Figura 30. Este comando hace referencia a la configuración automática tanto de MPLS como LDP, es decir que permitirá que los routers descubran y se comuniquen de forma automática con otros routers de la red mediante sesiones LDP y distribuir etiquetas.

Este tipo de configuración suele ser beneficios para redes grandes o en un entorno donde se requiere una implementación rápida y sin errores. Además, es

importante la verificación generada de forma automática ya que requiere ser controlada de forma adecuada.

Figura 30

Playbook con configuración MPLS



Realizado el programa, se procede a ejecuta el comando "*ansible-playbook -i hosts configuracion.yml*", ejecutando los comandos que se obtienen en la Figura 31, se habilita el MPLS y la autoconfiguración de LDP en los routers especificados en el inventario en este caso en los equipos P1, P2 y P3.

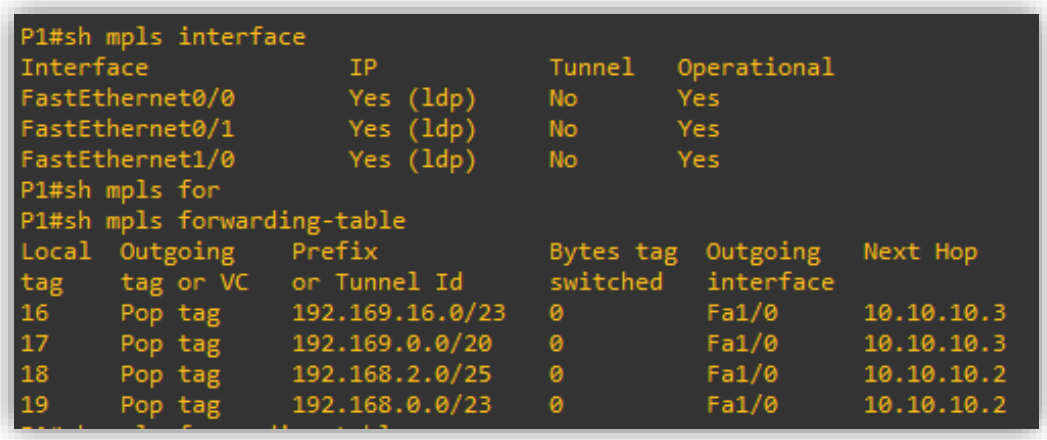
Con los comandos utilizados en la configuración manual que se muestran en la Figura 32, se puede visualizar como se está utilizando MPLS para reenviar paquetes dentro de la red, proporcionando información como resolución de problemas y monitoreo de rendimiento MPLS.

Figura 31

Configuración inicial en MPLS

Figura 32

Comandos para verificar el protocolo MPLS.



```
P1#sh mpls interface
Interface              IP            Tunnel   Operational
FastEthernet0/0        Yes (ldp)     No       Yes
FastEthernet0/1        Yes (ldp)     No       Yes
FastEthernet1/0        Yes (ldp)     No       Yes
P1#sh mpls for
P1#sh mpls forwarding-table
Local  Outgoing    Prefix           Bytes tag  Outgoing   Next Hop
tag    tag or VC   or Tunnel Id     switched   interface
16     Pop tag     192.169.16.0/23  0          Fa1/0      10.10.10.3
17     Pop tag     192.169.0.0/20   0          Fa1/0      10.10.10.3
18     Pop tag     192.168.2.0/25   0          Fa1/0      10.10.10.2
19     Pop tag     192.168.0.0/23   0          Fa1/0      10.10.10.2
```

4.3. Implementación de Políticas

La seguridad en una red ISP es uno de los aspectos importantes que se deben tener en cuenta al realizar en la implementación del diseño, debido a que nos permitirá proteger la información, los equipos, y aplicaciones que se encuentra en el sistema. Para que la seguridad sea correctamente aplicada se deben crear políticas de seguridad que permitan evitar posibles intrusiones.

La política de seguridad son los procedimientos que permiten establecer como se protegerá los activos de la red, los datos de los clientes, la infraestructura y los sistemas de la organización contra amenazas y riesgos de seguridad. Estas políticas garantizan la confidencialidad, integridad y disponibilidad de la información, así como para proteger la infraestructura de la red contra ataques maliciosos o incidentes de seguridad.

Por tal razón, para la creación de políticas de seguridad se utilizará un firewall Cisco ASA (Adaptive Security Appliance) que permita proteger la infraestructura de red y los datos de los clientes, mediante Ansible implica el uso de módulos específicos para interactuar con los dispositivos.

4.3.1. Zona desmilitarizada (DMZ)

La DMZ es una red intermediaria entre una red interna segura y externa no confiable, utilizada para alojar servicios que deben ser accesibles desde Internet sin tener acceso a la red segura. Las configuraciones básicas para la DMZ son: interfaces para las distintas zonas, NAT dinámico y estático, políticas de clase, rutas estáticas, mapas de clase, reglas indicadas por ACL (Padilla, 2023).

Para la configuración de la zona DMZ, se requieren las configuraciones siguientes para los interfaces como son: nivel de seguridad, la dirección IP, nombre de la interfaz, estado de la interface.

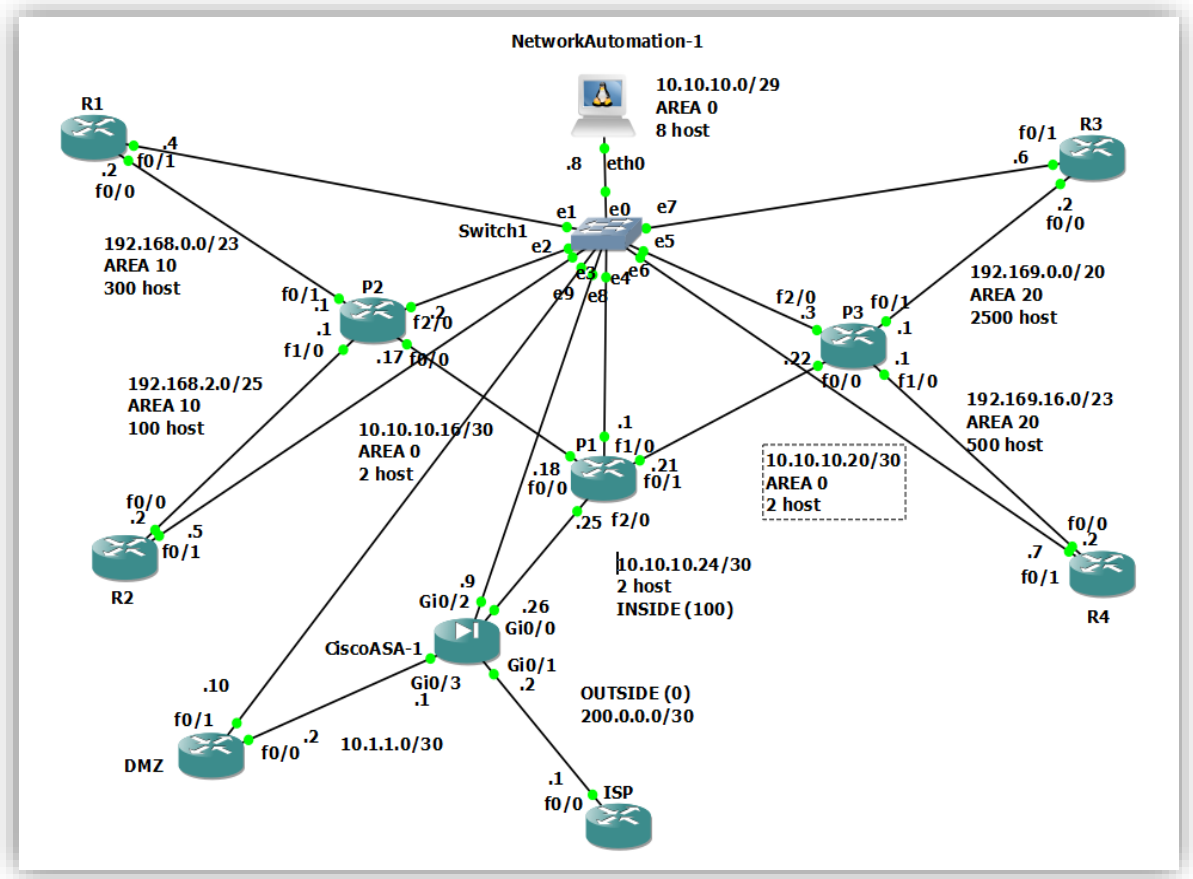
4.3.2. Configuración inicial

Inicialmente, se coloca el dispositivo ASA en el diseño ya implementado que se tiene en la Figura 33, en donde se tendrá una red INSIDE que será la red interna configurada con anterioridad, una red OUTSIDE será el ISP hacia el que está conectado, una red DMZ la cual utiliza para mejorar la seguridad de la red al aislar ciertos servicios y sistemas de la red interna y, al mismo tiempo, permitir que estos servicios sean accesibles desde el exterior de la organización.

Dado que el nivel de seguridad se utiliza para el control y administración del flujo de tráfico a través del dispositivo, para esto ASA ocupa un modelo mediante niveles de confianza numérico, es decir que va desde 0 (menos confiable) a 100 (más confiable). En la red se ha tomado en cuenta los siguientes niveles de seguridad para la red interna Inside se ha utilizado 100, porque los recursos deben ser confiables y seguros. En la red externa Outside y Ansible tendrá una red de 0, finalmente se ha dado un nivel de confianza de 50 a la red DMZ.

Figura 33

Diseño de la red automatiza con Ansible y políticas de seguridad con Cisco Firewall ASA

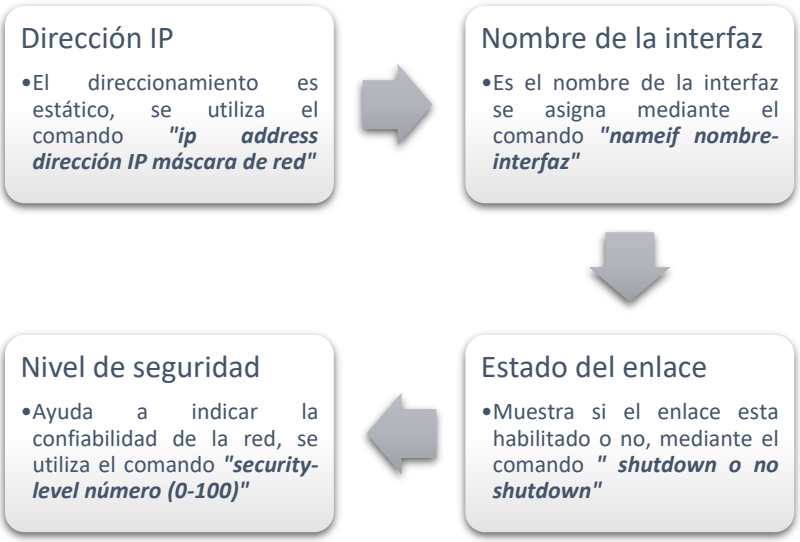


Hay que tener en cuenta que, al utilizar los niveles de seguridad, el tráfico se permite en el nivel más alto hacia el más bajo, pero no funcionaría de forma contraria. De esta manera, se controlará el tráfico que se permite configurando políticas de acceso mediante estos niveles.

Antes de lograr la ejecutar de las políticas de seguridad, se debe colocar la configuración de la interface del Firewall con Ansible de manera manual, para que se tenga conexión por ssh desde el mismo, en la Figura 34, se muestran los siguientes atributos son necesarios para la configuración.

Figura 34

Configuraciones a realizarse en los interaces del firewall ASA



Como se observa, la Figura 35, contiene la configuración de la interface Gi0/2 de la conexión con Ansible, se configura el direccionamiento IP, el nivel de seguridad, el nombre del enlace.

Figura 35

Configuración interfaces Cisco Firewall ASA

```
interface GigabitEthernet0/2
 nameif ANSIBLE
 security-level 0
 ip address 10.10.10.9 255.255.255.240
```

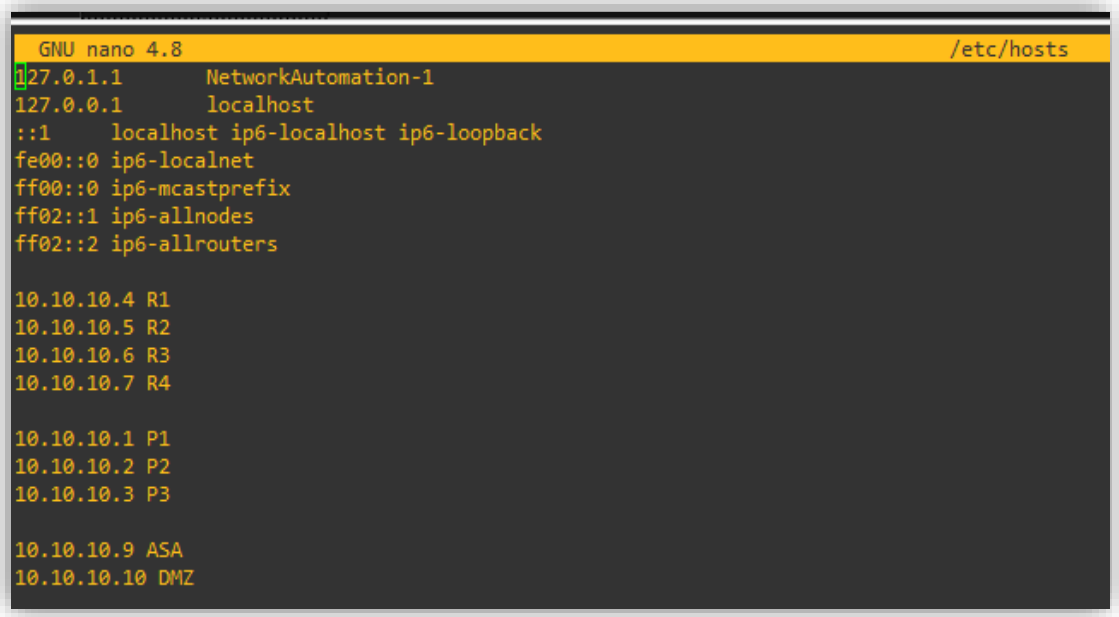
Sin embargo, para poder habilitar el acceso por ssh desde Ansible hacia el Firewall se debe colocar los comandos que se observan en la Figura 36, para que no exista ningún error al comunicarnos desde Ansible. El comando ***"enable password contraseña"*** es la contraseña para ingreso a enable, ***"username usuario password contraseña privilege 15"*** usuario y contraseña para el acceso ssh y el comando **"*sh key-exchange group dh-group14-sha1"*** utilizado para el intercambio de claves.

Figura 36

Configuración ssh en Cisco ASA

Figura 37

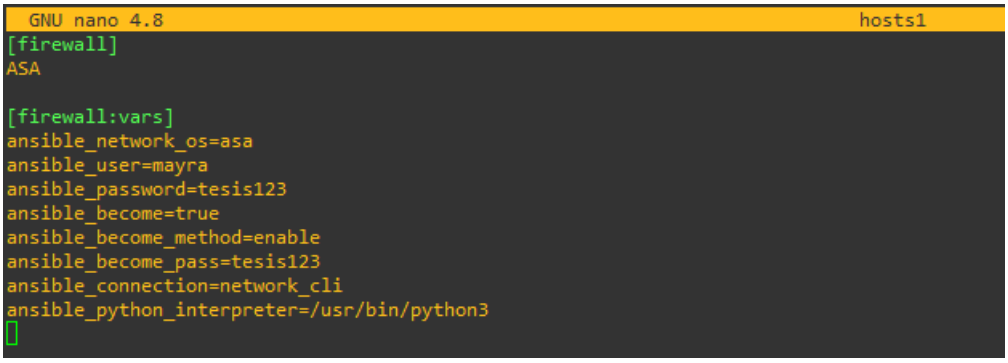
Agregar dirección IP para ASA y DMZ



Una vez realizada la configuración, se procede a colocar la dirección IP, del enlace que se tendrá comunicación con Firewall ASA, como se tiene en la Figura 37, en este caso ingresamos la dirección IP de Firewall y de DMZ, con su respectivo nombre, para su configuración automática. Los datos establecidos dentro de Ansible son 10.10.10.9 ASA y 10.10.10.10 DMZ.

Figura 38

Ingreso equipos en el inventario "host"



Por otra parte, estas direcciones IP, también se deben incluir en un archivo de inventario hosts llamado *"host1"*, para poder llamarlos desde el playbook, como se observa en la Figura 38.

La información agregada del firewall ASA en el repositorio es: nombre del usuario ***"ansible_host"*** en este caso el nombre es ASA, ***"ansible_network_os"*** el módulo de cisco.asa, ***"ansible_user y ansible_password"*** usuario y contraseña para SSH, ***"ansible_become"*** aceptación de ingreso de administración enable, ***"ansible_become_pass"*** es la contraseña enable, ***"ansible_connection y ansible_become_method"*** son las variables requeridas para la conexión, ***"ansible_python_interpreter"*** es el lenguaje de interpretación en este caso Python.

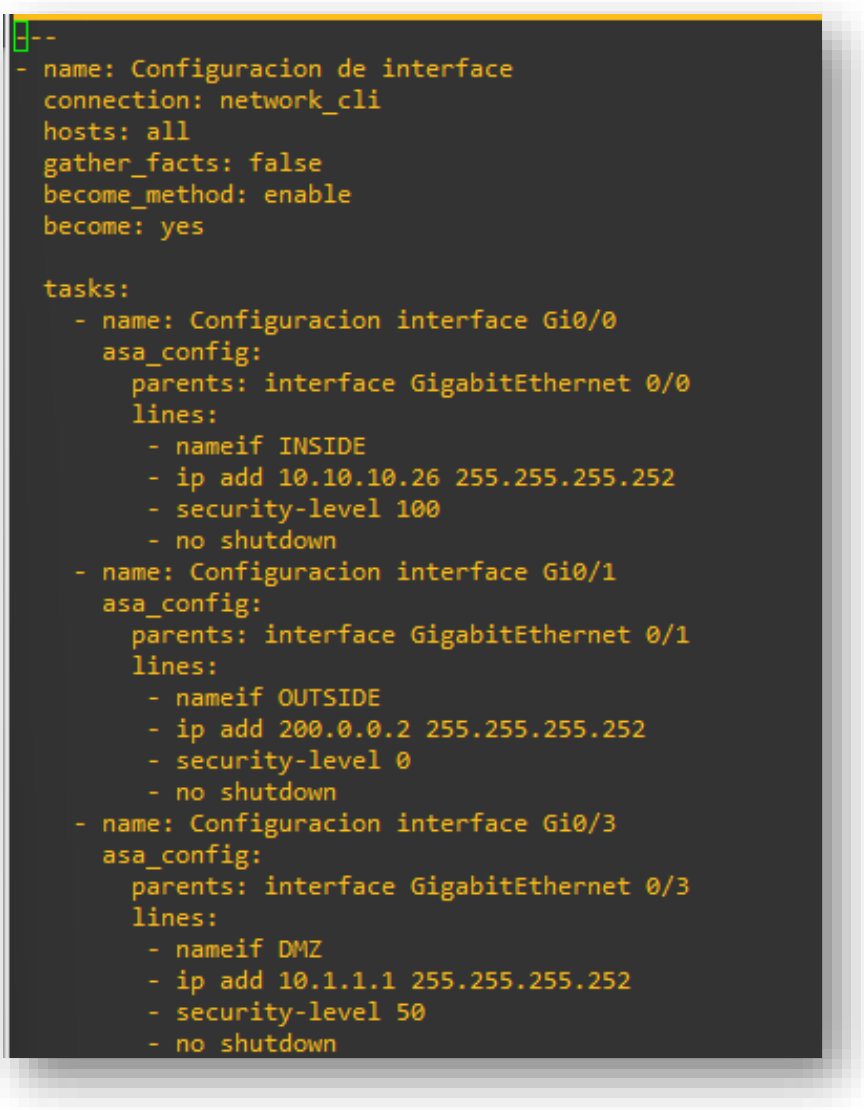
Para la configuración de las demás interfaces se puede realizar automáticamente mediante Ansible, como se presenta en la Figura 39, se realiza la programación en el playbook *"config_asa.yml"* en donde se tiene la configuración de los enlaces INSIDE, OUTSIDE, DMZ, colocando la dirección IP, el nombre de la interface, el nivel de seguridad que se le ha dado.

En este caso INSIDE se encuentra en la interface Gi0/0 con IP: 10.10.10.26 con un nivel de seguridad del 100% haciendo que la red sea segura. Por otra parte, OUTSIDE se encuentra en la interface Gi0/3 se tiene la dirección de 200.0.0.2, nivel de

seguridad 0. Finalmente se tendrá la red DMZ con dirección IP 10.10.10.1 y nivel de seguridad 50%.

Figura 39

Playbook de configuración de las interfaces y enrutamiento en Firewall ASA



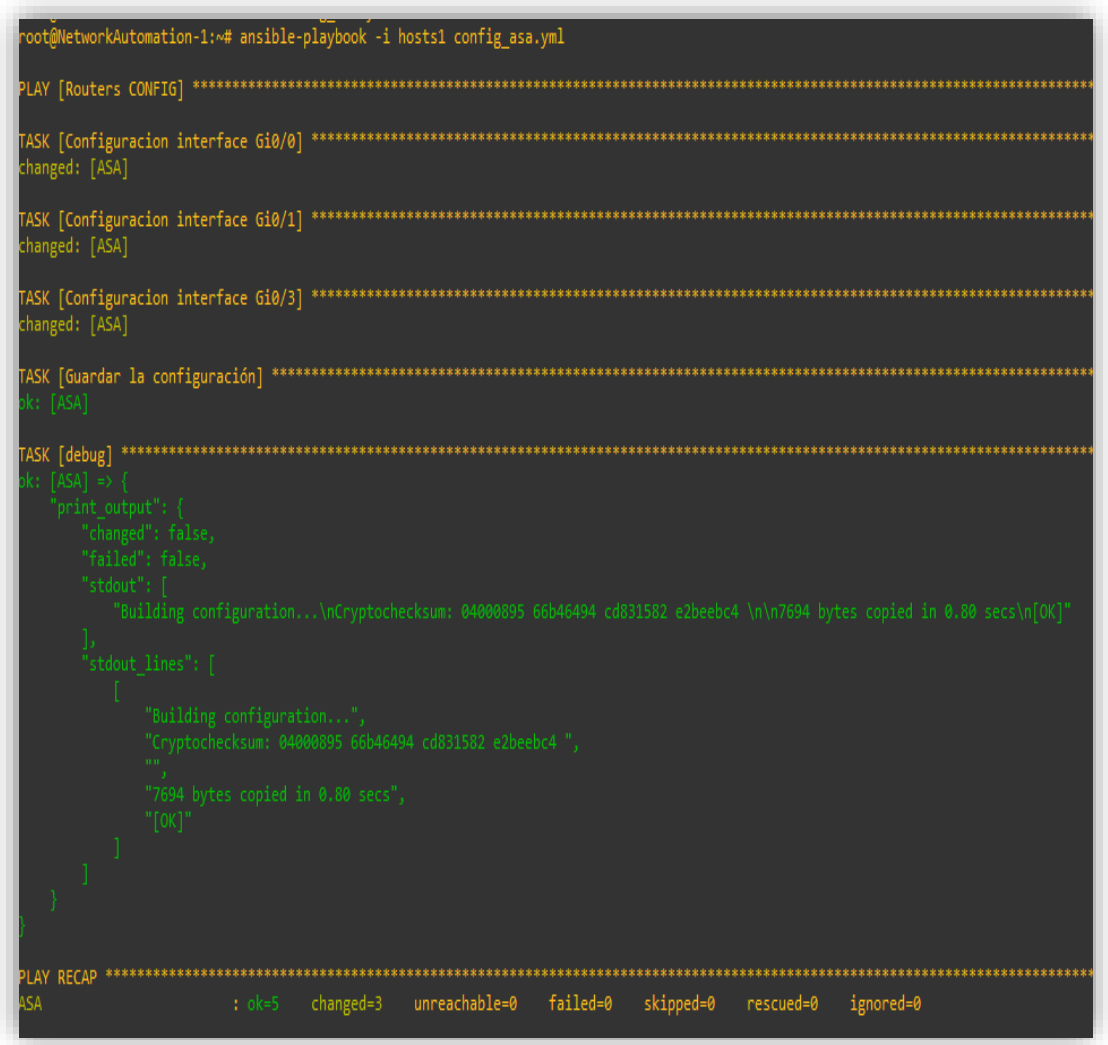
```
---
- name: Configuracion de interface
  connection: network_cli
  hosts: all
  gather_facts: false
  become_method: enable
  become: yes

  tasks:
    - name: Configuracion interface Gi0/0
      asa_config:
        parents: interface GigabitEthernet 0/0
        lines:
         - nameif INSIDE
         - ip add 10.10.10.26 255.255.255.252
         - security-level 100
         - no shutdown
    - name: Configuracion interface Gi0/1
      asa_config:
        parents: interface GigabitEthernet 0/1
        lines:
         - nameif OUTSIDE
         - ip add 200.0.0.2 255.255.255.252
         - security-level 0
         - no shutdown
    - name: Configuracion interface Gi0/3
      asa_config:
        parents: interface GigabitEthernet 0/3
        lines:
         - nameif DMZ
         - ip add 10.1.1.1 255.255.255.252
         - security-level 50
         - no shutdown
```

Por otra parte, en la Figura 40 se verifica la ejecución del comando playbook en ansible, en donde se muestra que fue realizado con éxito.

Figura 40

Configuración automática de las interfaces del Firewall ASA



Como siguiente paso se debe realizar la configuración del protocolo de enrutamiento, en este caso se utilizó el protocolo OSPF, en la Figura 41, se puede observar que las redes a aprender son 10.10.10.0 para la red Inside, 200.0.0.0 para la red Outside y la IP 10.1.1.0 para la conexión con el dispositivo DMZ.

Como se tiene en la Figura 42, se puede observar en el equipo Firewall que se han aprendido las siguientes antes mencionadas.

Figura 41

Configuración de la Cisco Firewall ASA mediante OSPF

```
- name: Habilitar OSPF
  asa_config:
    parents: router ospf 10
    lines:
      - network 10.10.10.0 255.255.255.0 area 0
      - network 200.0.0.0 255.255.255.252 area 0
      - network 10.1.1.1 255.255.255.252 area 0
- name: Guardar la configuracion
  asa_command:
    commands:
     - write
```

Figura 42

Rutas aprendidas en el Cisco Firewall ASA mediante OSPF.

```
ciscoasa# sh route ospf

Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, V - VPN
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, + - replicated route
Gateway of last resort is not set

D        10.10.10.16 255.255.255.252
           [110/20] via 10.10.10.25, 00:02:00, INSIDE
D        10.10.10.20 255.255.255.252
           [110/20] via 10.10.10.25, 00:02:01, INSIDE
D IA     192.168.0.0 255.255.254.0 [110/20] via 10.10.10.4, 00:01:52, ANSIBLE
                                   [110/20] via 10.10.10.2, 00:02:42, ANSIBLE
D IA     192.168.2.0 255.255.255.128
           [110/11] via 10.10.10.2, 00:02:42, ANSIBLE
D IA     192.169.0.0 255.255.240.0 [110/20] via 10.10.10.6, 00:02:22, ANSIBLE
                                   [110/20] via 10.10.10.3, 00:02:32, ANSIBLE
D IA     192.169.16.0 255.255.254.0 [110/11] via 10.10.10.3, 00:02:32, ANSIBLE
```

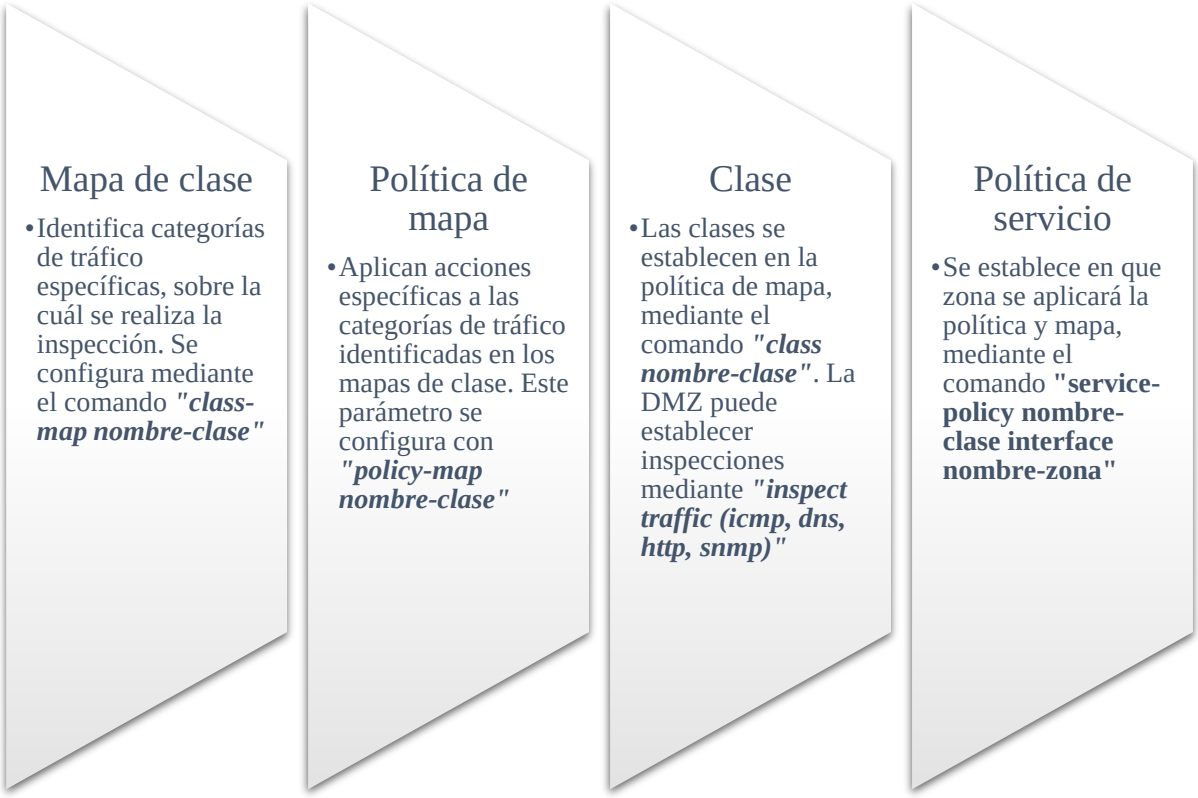
4.3.3. Configuraciones políticas y mapas

Las políticas y mapas de clase se utilizan para controlar y dar forma al tráfico de la red, aplicando las reglas de filtrado, priorizando el tráfico, entre otras acciones que se pueden realizar. Al implementar estas políticas permiten que un equipo pueda o no para acceder a una zona.

En la Figura 43, se muestra las diferentes configuraciones que se podría implementar en el programa, como el mapa de clase, política de mapa, clase y política de servicios.

Figura 43

Configuraciones de las políticas y mapas de clase.



En el playbook que se muestra en la Figura 44, se procede a realizar 2 tareas para las políticas que se implementará, en la primera tarea en enviará los comandos a la configuración global en donde se enviará las políticas de mapa y la clase. Después en la segunda tarea se indicará el tráfico a inspeccionar en este serán ICMP, SNMP y HTTP.

Una de las políticas que se va a implementar es el protocolo ICMP, que es utilizado para enviar mensajes de control y error dentro de una red IP, además de emplearlo para pruebas de conectividad. También se ha implementado la inspección de

SNMP Y HTTP. En este caso, esta política se crea de forma global bajo los comandos que se muestran en la Figura 45, se puede utilizar de forma específica en la interface.

Figura 44

Comando para crear política de seguridad.

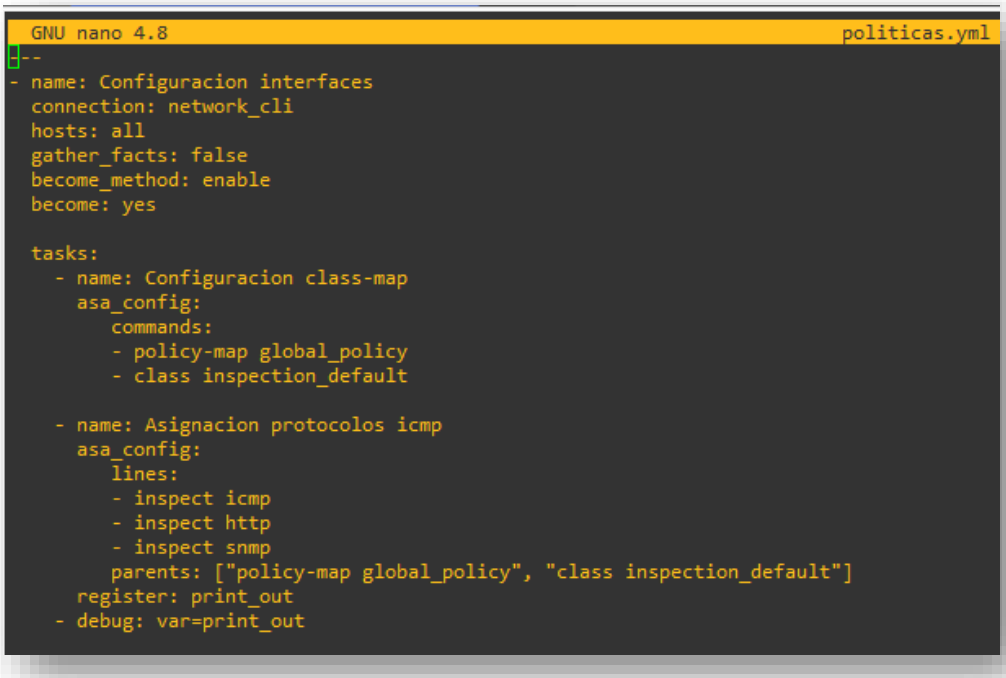
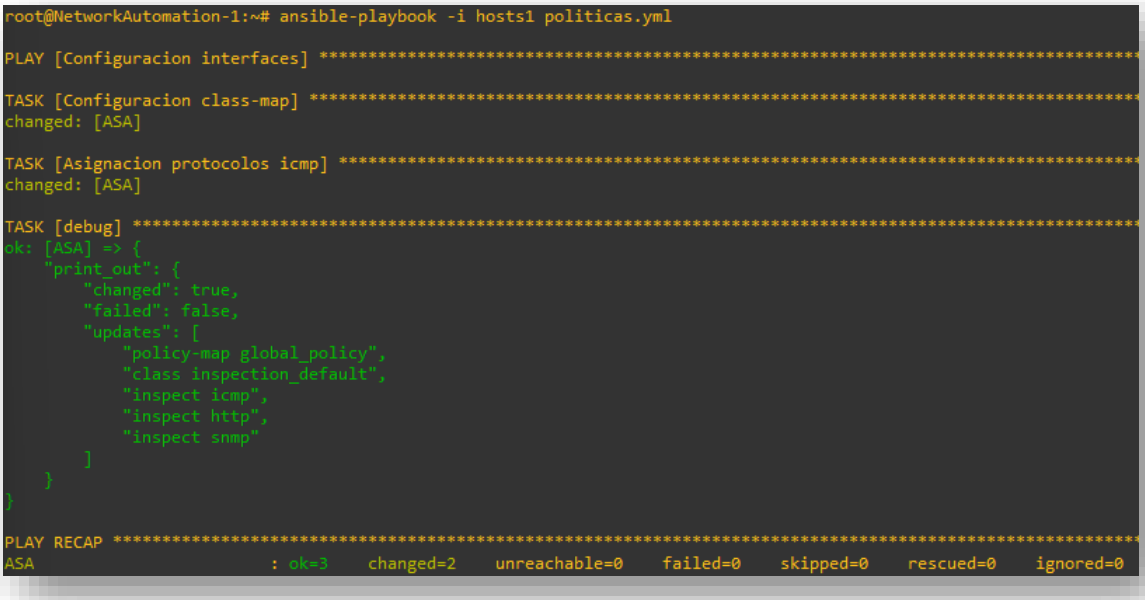


Figura 45

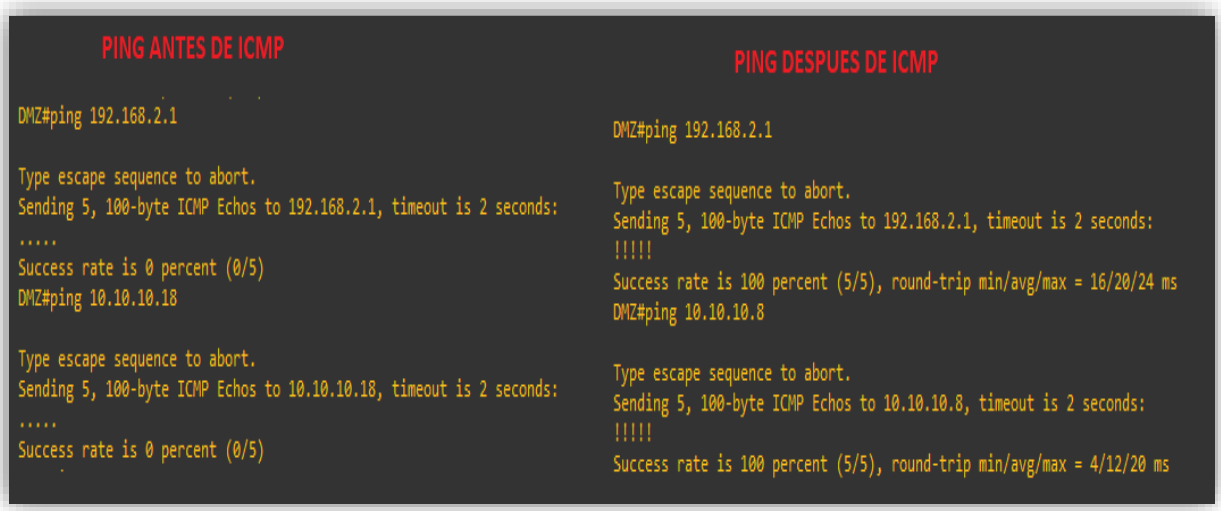
Compilación de playbook de las políticas de seguridad.



Como se observa en la Figura 46, antes de que se tenga implementado ICMP, no se contaba con ping desde el DMZ hacia la red INSIDE, debido a que se tiene restricciones el nivel de seguridad implementado. Posteriormente a la aplicación se puede observar el ping hacia la red interna, verificando que el protocolo funcionando correctamente.

Figura 46

Comprobación de protocolo ICMP en la red.

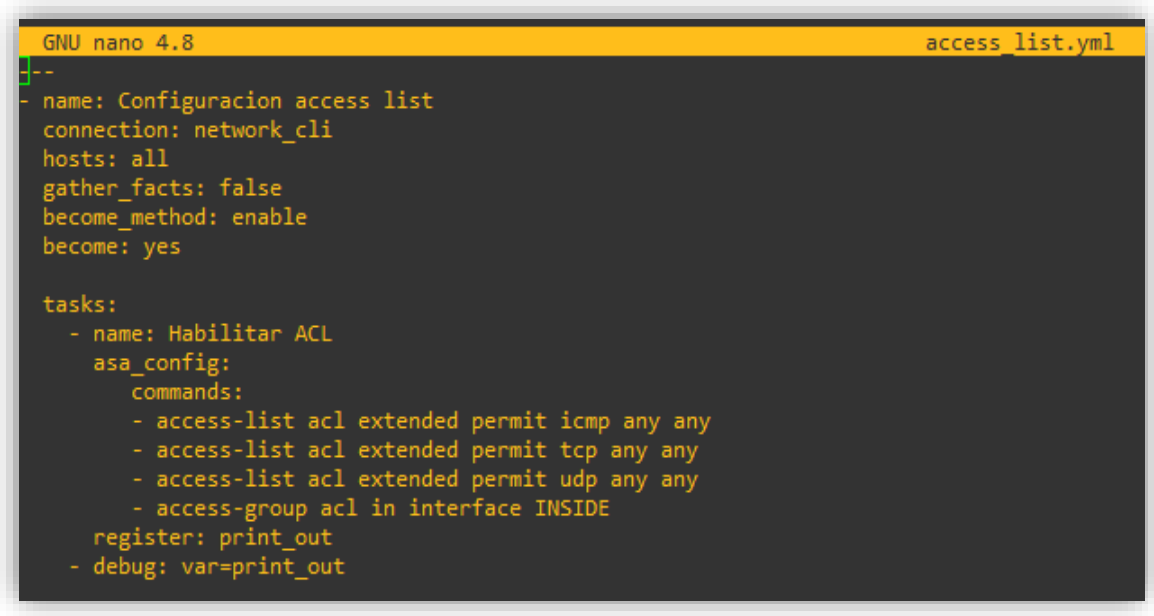


4.3.4. Configuración de access-list

A continuación, se muestra listas de acceso, que permitan controlar el flujo de tráfico basados en diferentes criterios, definidos mediante políticas de seguridad, para la protección contra amenazas y accesos no autorizados. En este caso se crea una lista de acceso de ICMP, UDP y TCP, como se tiene en la Figura 47. Es así, que se crea 3 ACL en el playbook.

Figura 47

Configuración de lista de acceso



CAPÍTULO 5
ANÁLISIS DE RESULTADOS

5.1. Comparación de red de forma manual y automática

Una vez finalizada las configuraciones totales de red, se procede a realizar un comparativo entre la configuración de manera manual y automática, para determinar las ventajas y desventajas que brinda cada una de las redes, y de esta manera diagnosticar que red es más eficiente y efectiva al momento de la configuración de los equipos.

Tabla 9

Ventajas de la configuración manual vs automática.

Configuración automática	Configuración manual
Implementa cambios en la red de forma rápida y eficiente en redes a gran escala.	Ofrece el control total sobre cada aspecto de la red.
Reduce de errores humanos, garantizando estabilidad en la red.	En situaciones particulares permite tomar decisiones personalizadas.
Facilita la gestión y expansión de redes grandes, sin necesidad de tanto personal técnico.	La configuración manual a menudo requiere un alto nivel de conocimiento técnico.
Permite mantener un control de cambios para identificar errores en la red de manera rápida.	En algunos casos, ofrece seguridad ya que no será posible la explotación de vulnerabilidades en scripts automáticos.

Al mostrar las ventajas de cada una de las configuraciones de red que muestra la Tabla 9, es necesario tener en cuenta que la red también depende de los recursos que el proveedor de servicio ISP brinda, así como, también los requerimientos de implementación de la red.

Entre las principales ventajas que se puede destacar de la configuración automática es la que permitirá trabajos de manera forma rápida, estable y segura la cual permite simplificar, la gestión y el manejo de la red. Sin embargo, existen casos

específicos donde requiere alguna configuración especial la misma que debe ser ejecutada de forma manual sin importar el tipo de servicio que mantenga la red.

Por otra parte, la Tabla 10 muestra las desventajas que cada una de las configuraciones presenta, destacando que la configuración automática mal implementada tendrá los mismos problemas que una configuración manual ya que esto obligará a encontrar el error y corregirlo de forma manual.

Tabla 10

Desventajas de la configuración manual vs automática.

Configuración automática	Configuración manual
Compleja, alto tiempo en desarrollo de scripts, herramientas y flujos de trabajo.	Propenso a errores humanos, que pueden ser difícil de detectar.
Errores en el script causa problemas graves en la red difíciles de diagnosticar.	Mayor tiempo de implementación, ralentizando las actualizaciones y cambios.
La actualización a versiones posteriores de la herramienta Ansible puede causar fallas por compatibilidad de aplicaciones utilizadas.	Dificulta el crecimiento de la red ya que requiere más tiempo y recursos.
La implementación de la herramienta de automatización requiere una inversión inicial, significativa.	Susceptible a errores de configuración ya que esta puede variar entre dispositivos.
	Complica mantener información documentada completa de la red.

Los diagramas que se tienen en el Anexo 2 y 3, ayuda a determinar los diferentes beneficios entre las configuraciones manual y automática de una red ISP, es necesario tener en cuenta los aspectos como la eficiencia, el tiempo de implementación, precisión, consistencia, escalabilidad, mantenimiento, actualizaciones de la red, seguridad y costos de implementación, serán la base para determinar la mejor configuración. La Tabla 11, muestra las variables a comparar y los valores a calificar en relación a su funcionalidad, siendo (3) el valor más alto y (1) el valor más bajo.

Tabla 11

Asignación de valores para los beneficios de la configuración manual y automática.

Eficiencia	(3) Muy Eficiente (2) Poco Eficiente (1) Nada Eficiente
Tiempo de implementación	Se agregará el tiempo de implementación en horas. (3) 1-5 horas (2) 6-10 horas (1) 11 a más horas
Precisión y consistencia	(3) alta precisión y consistencia (2) media precisión y consistencia (1) baja precisión y consistencia.
Escalabilidad	(3) redes grandes (2) redes medianas (1) redes pequeñas
Mantenimiento y actualizaciones	(3) fácil (2) regular (1) difícil
Seguridad	(3) Muy segura (2) Segura (1) Poco segura
Costos	(3) Mayor costo (2) Costo normal (1) Poco costo

En la tabla 12, se efectúa la calificación de cada aspecto tanto de forma manual como automática, con esta valoración se obtiene un gráfico de barras como se muestra en la Figura 39, el mismo que posibilita identificar la mejor configuración de red.

Uno de los beneficios las configuraciones, es la eficiencia que será mayor al utilizar Ansible porque puede automatizar tareas repetitivas y desplegarla en múltiples dispositivos de manera simultánea a comparación de la configuración manual, ya que cuando se tienen redes grandes aumenta la cantidad de dispositivos. Por otra parte, la configuración automática ofrece una alta precisión y consistencia debido a que sigue los mismos procesos de ejecución, disminuyendo la probabilidad de errores humanos.

Tabla 12

Calificación para los aspectos de las configuraciones manual y automático.

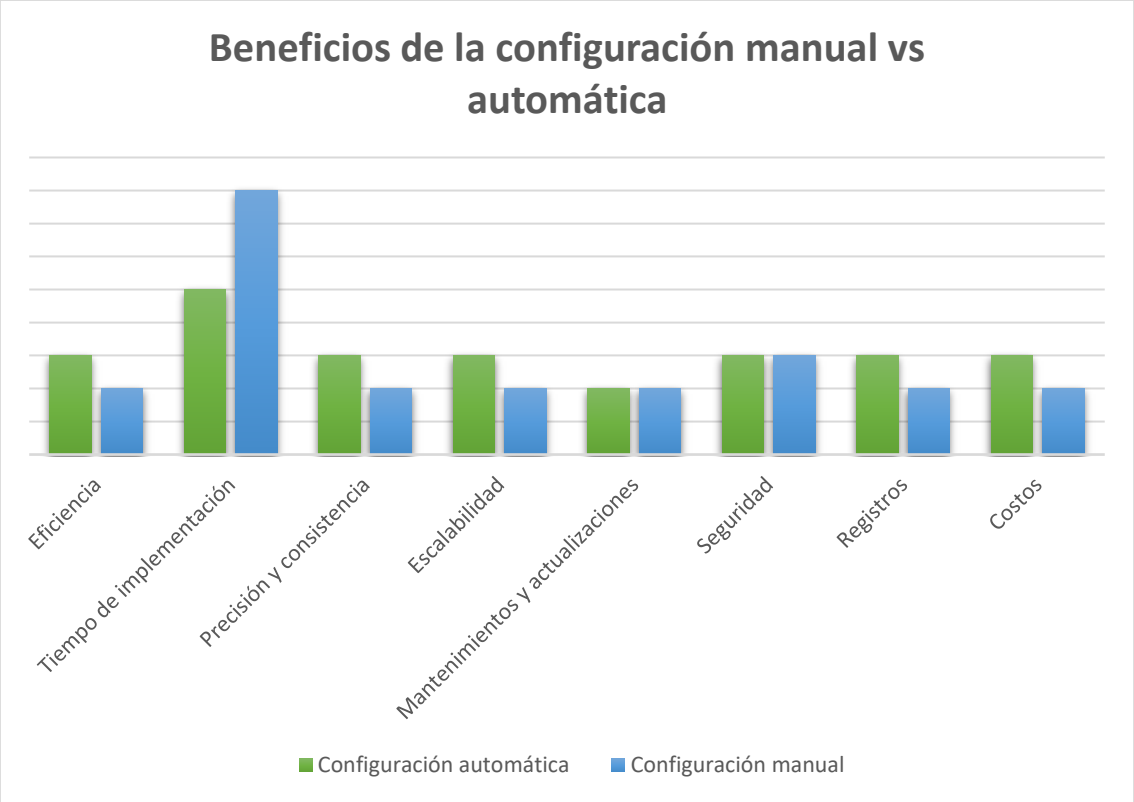
	Configuración automática	Configuración manual
Eficiencia	3	2
Tiempo de implementación	3	2
Precisión y consistencia	3	2
Escalabilidad	3	2
Mantenimiento y actualizaciones	3	3
Seguridad	2	2
Costos	3	2

Además, para la escalabilidad será más fácil la configuración con Ansible en el caso de que se requiera configurar nuevos dispositivos y se tendrá un tiempo menor que en una configuración manual. En caso de requerir un mantenimiento o actualización, este, reduce el riesgo de interrupciones no planificadas. Por otra parte, cuando tiene que ver con la seguridad dependerá de cómo se realice la configuración de red por parte del administrado, sin embargo, de forma automática, facilita el seguimiento y registro de cambios en la configuración. Finalmente, en cuanto al costo se va a requerir una mayor inversión inicial al momento de implementar una configuración automática, sin embargo, una vez instalada y adquirida su costo de mantenimiento será menor que el mantener una configuración tradicional, no obstante, en la configuración tradicional puede existir mayor costo en la mano de obra a medida que la red crece ya que es necesario contratar más técnicos especializados que realicen el trabajo.

Esto implica que la configuración automática con Ansible ha generado mayores ventajas al momento de su aplicación a comparación de la configuración manual, es necesario mencionar que esto dependerá de la red ISP que se implementará, además de otros factores como la escalabilidad red.

Figura 48

Beneficios de la configuración manual vs automática.



5.2. Indicadores de desempeño KPI

El establecer indicadores de desempeño permite verificar el funcionamiento de la configuración de red en la que la seguridad y la autonomía de la red ISP son fundamentales para determinar la importancia de monitorear la efectividad de la infraestructura y la resiliencia de la red. Por esta razón a continuación se muestra los KPIs recomendados a implementar en este tipo de redes.

5.2.1. Número de reglas firewall:

Se tomará en cuenta este aspecto como KPI para controlar en el caso de que exista un aumento, se requeriría la revisión y consolidación para mejorar la seguridad. El número de reglas de firewall es un indicador importante para evaluar la complejidad y el nivel de seguridad de una red automatizada.

Cuantas más reglas existan, mayor es la complejidad y el potencial de errores. Sin embargo, también es importante tener en cuenta que un número extremadamente bajo de reglas podría indicar una falta de seguridad.

Para definir el objetivo de este KPI se requiere que las políticas de seguridad se mantengan efectivas y simples. Se debe definir la métrica que se utilizará como, por ejemplo, el número total de regla, el número de reglas permitidas o negadas, entre otras, además de considerar como parámetro el tiempo si es en tiempo real o a largo plazo.

Posteriormente, se puede establecer una meta, por ejemplo, no tener más de 50 reglas dependiendo de red que se tenga. Entonces, puede automatizar la red para obtener estos valores. Con estos valores obtenidos se puede realizar un análisis que verificar si es innecesaria la política.

5.2.2. Detección y prevención de intrusiones:

Permite la supervisión de la cantidad de intrusiones detectadas y bloqueadas. El objetivo será mejorar la eficacia de la detección de amenazas o reducción de incidentes de seguridad. Después se deber indicar cual es la métrica de la elección, en este caso puede ser de ayuda el número de incidente de seguridad, el tiempo de detección o tiempo de respuesta, falsos positivos, entre otros.

Obtenido los datos, se puede analizar esta información para identificar tendencia a largo plazo, como el aumento o disminución de incidente de seguridad en la red. Para la detección de amenazas se requerirá un proceso continuo de monitorización y ajustando medidas de seguridad para adaptarse a amenazas.

5.2.3. Tráfico Denegado por regla de firewall:

mide la cantidad de tráfico que se ha bloqueado debido a las reglas de firewall. Es importante para evaluar el rendimiento y la eficacia de las políticas de seguridad implementadas. La métrica que se podría utiliza es el número de conexiones o sesiones de tráfico negado. En este caso se podría establecer una meta de reducir el número de conexiones de tráfico denegado en un 10% en el próximo mes.

Si las métricas implementadas no cumplen, se considera tomar medidas para mejorar la seguridad y reducir el tráfico denegado. Esto podría implicar ajustar las reglas de firewall, implementar nuevas soluciones de seguridad, actualizar firmas de amenazas, etc.

5.2.4. Disponibilidad de la red:

Monitorear el tiempo de actividad de la red, garantizando que la red y los servicios estén disponibles en todo momento. En este caso se puede utilizar métricas como: tiempo de inactividad, tiempo medio entre fallos o reparación.

Posteriormente se puede tener una meta a alcanzar de un 99.9% de disponibilidad durante la red. Para esto se puede automatizar la recopilación de registros de eventos, registros de disponibilidad y estadísticas de rendimiento de tu infraestructura de red y firewall ASA.

Después, se compara las métricas actuales con tus objetivos y detecta patrones de tiempo de inactividad, MTBF, MTTR, entre otros. Si tus métricas no cumplen con tus objetivos o límites de disponibilidad, considera tomar medidas para mejorar la disponibilidad. Esto podría implicar mejorar la redundancia de la infraestructura,

implementar medidas de seguridad adicionales o fortalecer las políticas de recuperación ante desastres.

La disponibilidad es un objetivo continuo, que se debe monitorear y ajustar la infraestructura y políticas de seguridad de manera regular para mantener la disponibilidad y la resistencia de tu red.

5.2.5. Escalabilidad:

Medir la capacidad de la red para manejar un aumento de carga de tráfico sin degradación del servicio. Define el objetivo KPI, puede ser el asegurarse de que tu red y políticas de seguridad puedan expandirse para admitir nuevos servicios o usuarios sin degradación del rendimiento.

Se puede utilizar métricas como capacidad de la red para manejar el tráfico y las conexiones, también se puede tener en cuenta el tiempo de respuesta. Una de las metas a aplicar será que la red es capaza de manejar el doble de usuario sin degradación.

Backups y restauraciones:

Se realiza copias de seguridad regulares de la configuración de los dispositivos y mide el tiempo necesario para restaurar la configuración en caso de problemas. Se puede evaluar la efectividad de las medidas de respaldo y recuperación de datos.

Un parámetro a considerar es asegurarnos de que los datos y la configuración esten respaldados, y serán restaurados cuando sea el caso de pérdida o desastre. Un ejemplo será el tener una meta de realizar copias de seguridad diarias y restaurar datos en menos de una hora en caso de pérdida.

Ansible puede ayudarte a automatizar el proceso de backup y restauración, así como a generar registros de actividad. Analiza los datos recopilados para evaluar si

estás cumpliendo tus metas y límites relacionados con el backup y restauración. Compara las métricas actuales con tus objetivos y detecta patrones de tiempo de backup, tiempo de restauración, etc.

CAPÍTULO 6

CONCLUSIONES Y RECOMENDACIONES

6.1. CONCLUSIONES

Mediante el diseño de la red utilizando configuración automatizada con la plataforma Ansible y la implementación de políticas de seguridad en el dispositivo Firewall Cisco ASA, es una estrategia efectiva para optimizar la gestión de la configuración, además de garantizar la seguridad de la red. Ofreciendo una mayor agilidad, administrando la red forma eficiente contrarrestando amenazas.

Como resultado de la automatización de configuraciones, se tiene una solución estratégica y efectiva para mantener y proteger una red siempre y cuando este en constante crecimiento, al tiempo que reduce los costos operativos. Sin embargo, es importante destacar que la implementación y el mantenimiento exitosos de esta solución requieren una planificación cuidadosa y una gestión continua, dependiendo de la red a utilizarse.

Al realizar la evaluación del funcionamiento de la simulación y su diseño con la combinación de automatización y las políticas de seguridad han demostrado una red más confiable y resistente a amenazas.

Al culminar el diseño de la red se determinó diferentes KPIs que permiten evaluar parámetros como el número de reglas de firewall, la escalabilidad, disponibilidad de la red, trafico denegado por reglas de firewall, los cuales garantizarán la continuidad del servicio y de esta formar disminuir el riego de amenazas.

6.2. RECOMENDACIONES

Para el diseño de la topología de la red es importante verificar que exista redundancia y sea escalable para adaptarse a un crecimiento futuro. Además, se debe analizar varios factores para determinar si lo adecuado es implementar la automatización de las configuraciones de la red.

Las políticas de seguridad deben ser sólidas en toda la red, para garantizar que sea efectiva, esto quiere decir que se deberá utilizar lista de control de accesos, sistemas de detección de intrusiones.

Con los KPIs determinados es recomendable el monitoreo y registro continuo en el sistema para detectar y responder rápidamente a amenazas o comportamientos no deseados en la red.

CAPÍTULO 7
REFERENCIAS

Agapidis, L. (2023). *Comparison of GNS3 vs EVE-NG vs Packet Tracer for Networks Simulation.* Obtenido de Networks Training: https://www.networkstraining.com/gns3-vs-eve-ng-vs-cisco-packet-tracer/

Ángulo, J., & Camacho, P. (2006). *Protocolos de enrutamiento* . Universidad Tecnológica de Bolivar.

Bolaños, V. (2023). *Configuración de OSPF con Huawei: Dos Áreas*. Obtenido de https://www.lo0.es/configuracion-de-ospf-dos-areas-con-huawei/

Brikman, Y. (2016). *Why we use Terraform and not Chef, Puppet, Ansible, Pulumi, or CloudFormation*. Obtenido de https://lsi.vc.ehu.eus/pablogn/docencia/AS/Act7%20Admin.%20centralizada/Terraform%20Chef%20Puppet%20Ansible%20Salt.pdf

Cisco. (2023). *Ejemplo de Conexión de Tres Redes Internas con Configuración de Internet.* . Obtenido de Cisco: https://www.cisco.com/c/es_mx/support/docs/security/asa-5500-x-series-next-generation-firewalls/113041-asa-3net.html

Cisco. (2023). *Firewalls de última generación Cisco ASA serie 5500-X*. Obtenido de https://www.cisco.com/c/es_es/products/security/asa-5500-series-next-generation-firewalls/index.html

Conde, G. (2021). *Automatización de redes informáticas con Python.* Universitat Oberta de Catalunya.

De Ghein, L. (2016). *MPLS Fundamentals: MPLS Fundamentals ePub _1*. Cisco Press.

Diéguez, L. (29 de 06 de 2020). *Tutorial Ansible desde 0 – Herramienta de gestión de servidores*. Obtenido de https://luisiblogdeinformatica.com/tutorial-ansible-desde-0-herramienta-de-gestion-de-servidores/

Díez Álvarez, J. (2016). *Despliegue de una red IP/MPLS para un ISP*. Madrid: Universidad Carlos III de Madrid. Departamento de Ingeniería Telemática.

EDUCBA. (2023). *Ansible Architecture | Simple Architecture of Ansible.* https://www.educba.com/ansible-architecture/.

Enciso, L., & Morales, C. (2021). *Ansible una estrategia de administración y configuración automatizada sobre GNS3 con OSPF para redes empresariales medianas.* Loja: Universidad Técnica Particular de Loja.

Entel. (2023). *Redes Privada MPLS.* Obtenido de https://www.entel.cl/corporaciones/fija/redes-privadas-mpls/

Figueroa, J., Rodriguez, R., Bone, C., & Saltos, J. (15 de 12 de 2017). *La seguridad informática y la seguridad de la información.* Obtenido de Polo del conocimiento: https://polodelconocimiento.com/ojs/index.php/es/article/view/420/pdf

Flórez, F. (2019). *Guía para el análisis del protocolo de encaminamiento OSPF en redes inalámbricas aplicado a nuevas tecnologías.* Bogotá: Universidad Santo Tomás de Colombia. .

Gil Martinez, J. L. (2015 de 2015). *Análisis y estudio de OSPF como protocolo de enrutamiento interno de una red corporativa.* Madrid: E.T.S.I. de Sistemas Informáticos (UPM).

Guevara, J. (2010). *MPLS, GMPLS, ASON*. Obtenido de https://www.ccapitalia.net/descarga/teleco/2010-guevara-mplsgmplsason.pdf

Huawei. (18 de 12 de 2022). *Conoce sobre la diferencias entre EIGRP y OSPF*. Obtenido de Huawei: https://forum.huawei.com/enterprise/es/Conoce-sobre-la-diferencias-entre-EIGRP-y-OSPF/thread/667235084119457792-667212882523336704

IBM Corporation. (14 de 04 de 2021). *OSPF (Open Shortest Path First)*. Obtenido de https://www.ibm.com/docs/es/i/7.2?topic=routing-open-shortest-path-first

IPCISCO. (2022). *Puppet Software | Network Automation Tool | How Puppet Works ⋆ IpCisco.* Obtenido de https://ipcisco.com/lesson/puppet-overview/

Mier Ruiz, E. E., & Mier Ruiz, G. D. (2008). *Protocolos de enrutamiento RIP, OSPF Y EIGRP*. Cartagena: Universidad Tecnológica de Bolivar.

Mier, E., & Mier, G. (2008). *Protocolos de enrutamiento RIP, OSPF y EIGRP*. Cartagena: Universidad Tecnológica de Bolivar.

Muñoz-Gallego, A. (2018). *Protocolos RIP y OSPF. Arquitectura de Redes y Servicios.* Obtenido de https://riuma.uma.es/xmlui/bitstream/handle/10630/16542/RIP%20y%20OSPF.pdf

Pacheco, X. (2022). *Análisis de herramientas de software para la simulación y emulación de redes de computadores para aprendizaje en entornos de laboratorio.* Machala: Universidad técnica de Machala.

Padilla, L. (2023). *Creación de DMZ en equipos especializados de red mediante Devops.* Quito: Escuela Poliécnica Nacional.

Progress Software Corporation. (2023). *Progress Chef*. Obtenido de https://www.chef.io/products/chef-infra

Puppet. (2023). *IT Process Automation and Task Orchestration.* Obtenido de https://www.puppet.com/why-puppet/use-cases/it-process-automation

Rajesh, K. (2022). *What is Chef and How it works? An Overview and Its Use Cases.* Obtenido de https://www.devopsschool.com/blog/what-is-chef-and-how-it-works-an-overview-and-its-use-cases/

Red Hat. (2023). *Red Hat Ansible Automation Platform*. Obtenido de Red Hat: https://www.redhat.com/en/technologies/management/ansible

Red Hat Inc. (2023). *Automatización de redes para todas*. Obtenido de https://www.redhat.com/es/engage/network-automation-everyone-s-202101221234

Red Hat Inc. (2023). *What is an Ansible Playbook?* Obtenido de https://www.redhat.com/en/topics/automation/what-is-an-ansible-playbook#:~:text=Ansible%20Playbooks%20are%20lists%20of,in%20which%20they%20are%20written.

Rios, I. (2019). *Key Performance Indicators (KPI)*. Obtenido de https://gc.scalahed.com/recursos/files/r161r/w24174w/S8_desarrollo_aplicacion_gestion.pdf

Telecapp Inc. (2023). *Protocolos de enrutamiento.* Obtenido de Telecapp: https://telecapp.com/protocolos-enrutamiento

Valverde, B. (2016). *Análisis comparativo de los protocolos de enrutamiento dinámicode una red de transporte para la Universidad de Guayaquil.* Guayaquil: Universidad de Guayaquil.

VMware. (2023). *Arquitectura del sistema de Salt*. Obtenido de WMware: https://docs.vmware.com/es/VMware-vRealize-Automation-SaltStack-Config/8.6/use-manage-saltstack-config/GUID-8FC70D95-3317-4324-A5BD-D213CE9B029E.html

Wågbrant, S., & Dahlén, V. (2022). *Automated Network Configuration a Comparison Between Ansible, Puppet, and SaltStack for Network Configuration.* Mälardalen University.

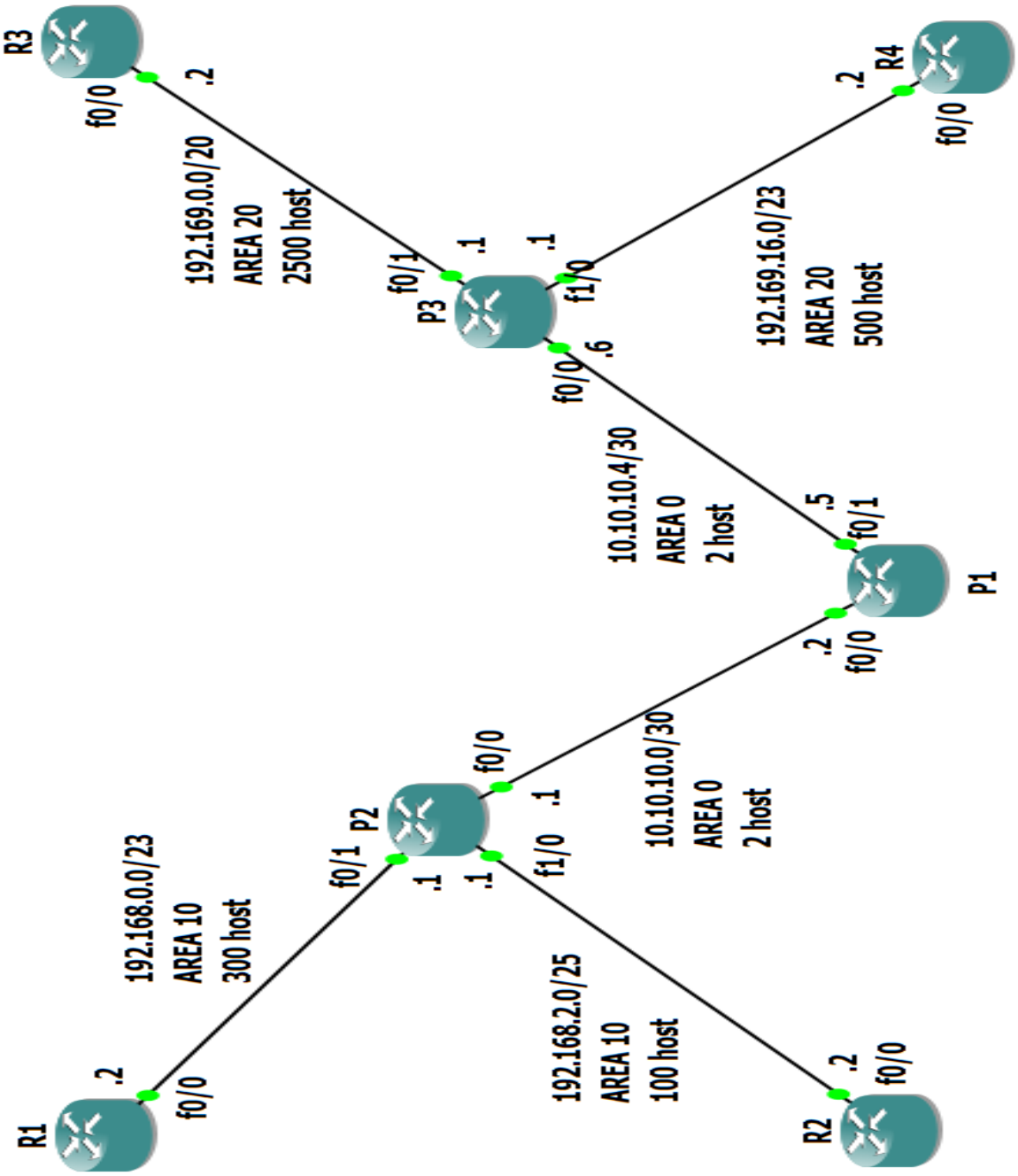
Yunga, A. (2018). *Implementación del provisionamiento automático de configuraciones (Network Automation) en infraestructura multivendor con Ansible.* Chimborazo: Escuela Superior Politécnica de Chimborazo.

CAPÍTULO 8

ANEXOS

ANEXOS 1

Diagrama de la red ISP configurado de manera manual



ANEXOS 2

Diagrama de red ISP con configuración automática.