

**manual técnico**

**SISPRO**

Versión 1.0 Agosto 2018

ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL

CONMUTACIÓN Y ENRUTAMIENTO

Contents

[**1.** **INTRODUCCIÓN** 4](#_Toc522780252)

[**2.** **RECURSOS DE SOFTWARE Y SIMULACIÓN** 4](#_Toc522780253)

[**2.1.** **Java v8.181** 4](#_Toc522780254)

[**2.2.** **Java.sql** 4](#_Toc522780255)

[**2.3.** **JavaFX Scene Builder** 4](#_Toc522780256)

[**2.4.** **Scene Builder** 5](#_Toc522780257)

[**2.5.** **GNS3** 5](#_Toc522780258)

[**3.** **RECURSOS DE SIMULACIÓN** 5](#_Toc522780259)

[4. RECURSOS DE HARDWARE 5](#_Toc522780260)

[4.1. Switch Catalyst 2960 5](#_Toc522780261)

[4.2. Enrutador cisco 2811 5](#_Toc522780262)

[**5.** **ANTECEDENTES Y CASO DE ESTUDIO** 6](#_Toc522780263)

[5.1. Diagrama de red 6](#_Toc522780264)

[5.2. Direccionamiento IP 7](#_Toc522780265)

[5.3. Definición de la arquitectura del sistema. 8](#_Toc522780266)

[**6.** **CONFIGURACIONES DE LOS DIPOSITIVOS DE RED** 8](#_Toc522780267)

[6.1. Plantilla Básica 8](#_Toc522780268)

[6.2. Configuración de un Provider 9](#_Toc522780269)

[6.2.1. Configuración de direccionamiento 9](#_Toc522780270)

[6.2.2. Configuración de IGP para comunicación entre Provider Edges. 10](#_Toc522780271)

[6.2.3. Configuración de MPLS y LDP de forma global. 10](#_Toc522780272)

[6.2.4. Configuración de Customer Edges. 10](#_Toc522780273)

[6.3. Configuración de Provider Edges. 11](#_Toc522780274)

[6.3.1. Configuración de direccionamiento para los PE. 11](#_Toc522780275)

[6.3.2. Configuración de IGP 12](#_Toc522780276)

[6.3.3. Comunicación entre Provider Edges. 12](#_Toc522780277)

[6.3.4. Activación de protocolo de distribución de etiquetas. 12](#_Toc522780278)

[6.3.5. Permitir que los enrutadores propaguen información mediante el protocolo MP-BGP. 13](#_Toc522780279)

[6.3.6. Creación de tablas de enrutamiento virtuales. 13](#_Toc522780280)

[6.3.7. Establecer comunicación entre dispositivos PE y CE. 14](#_Toc522780281)

[6.3.8. Redistribución de rutas con el protocolo MP-BGP 15](#_Toc522780282)

[**7.** **Código Fuente.** 15](#_Toc522780283)

# **INTRODUCCIÓN**

SISPRO es un aplicativo de escritorio que permite realizar configuraciones a dispositivos Provider Edge mediante un acceso remoto. Está diseñado especialmente para configurar dispositivos intermediarios pertenecientes al backbone de una red MPLS de un ISP. Estas configuraciones involucran temas como creación de VLANs, de tablas de enrutamiento virtuales pertenecientes a los enrutadores Provider Edge, y la asignación de direcciones IP disponibles para instalar un cliente en la red.

**Observación:** De ahora en adelante nos referiremos a los dispositivos Provider Edge y Customer Edge como PE y CE respectivmente. Para un mayor entendimiento se recomienda consultar terminología MPLS o acudir al glosario al final del manual.

**Nota:** SISPRO está dedicado solamente a realizar configuraciones dentro de los enrutadores PE. Aspectos como configuración básica o configuración de dispositivos CE, no son correspondidos.

* 1. Funciones del sistema
* Permitir la gestión de direccionamiento IP a personal autorizado.
* Crear redes virtuales de área local y tablas virtuales de enrutamiento para los PE.
* Automatizar la configuración de los dispositivos del área del proveedor, es decir los Provider Edges.
* Proveer una interfaz gráfica de fácil interpretación para la configuración de los dispositivos.

# **RECURSOS DE SOFTWARE Y SIMULACIÓN**

A continuación, especificamos los requerimientos de software y librerías a utilizar.

## **Java v8.181**

Java es un lenguaje de programación y una plataforma informática creada por Sun Microsystems. Hay muchos programas y sitios web que dependen su funcionamiento en que se tenga Java instalado. Java es rápido, seguro y fiable. Escogemos java como solución mejor cualificada para este panorama de la programación orientada a objetos.

## **Java.sql**

Provee APIs para acceder y procesar a los datos almacenadas en las bases SQL, por medio del lenguaje de programación de Java.

## **JavaFX Scene Builder**

Esta disponibilidad de JavaFX permite trabajar con el software Scene Builder proporcionando facilidades para relacionarlo con su código fuente.

## **Scene Builder**

Este entorno de trabajo permite el diseño de interfaces de usuario de una manera sencilla como lo es “arrastrar y soltar”, y trabaja con el ambiente de JavaFX, lo que permite asignar eventos a los objetos creados mediante este software de una forma fluida.

## **GNS3**

GNS3 es un simulador gráfico de red que permite el diseño de topologías de redes de computadores de forma avanzada, y realizar simulaciones entre ellos estableciendo parámetros muy cercanos a la realidad.

Para permitir completar simulaciones, GNS3 está estrechamente vinculada con:

* Dynamips, un emulador de IOS que permite a los usuarios ejecutar imágenes binarias IOS de Cisco System.
* Dynagen, un front-end basado en texto para Dynamips
* Qemu y VirtualBox, para permitir utilizar máquinas virtuales como un firewall PIX.
* VPCS, un emulador de PC con funciones básicas de networking
* IOU (IOS on Unix), compilaciones especiales de IOS provistas por Cisco para correr directamente en sistemas UNIX y derivados.

GNS3 es una excelente herramienta complementaria para el estudio de las tecnologías de redes, debido a su potencial virtualizando hardware de dispositivos reales.

# **RECURSOS DE SIMULACIÓN**

Las pruebas de funcionalidad de SISPRO en su etapa de desarrollo fueron realizadas bajo el software ***GNS3,*** el cual permite la simulación de sistemas de redes virtualizando componentes y dispositivos de red reales permitiendo a los usuarios aprovechar diferentes entornos de simulación.

El IOS usado para simular los enrutadores corresponde a *c7200-adventerprisek9-mz.153-3.XB12* el cual representa a los enrutadores 7200 de Cisco.

**Importante:** Esta versión de IOS de los enrutadores no cuentan con *Modo Usuario,* por esto hay que tomar correctivos en el código al momento de usar otra versión del IOS.

# RECURSOS DE HARDWARE

## **Switch Catalyst 2960**

* Soportan voz, video, datos y acceso seguro.
* Disponen de 24 puertos 10/100 más dos puertos SFP.
* Capacidad de configurar LAN virtuales.
* Seguridad integrada.

## **Enrutador cisco 2811**

* Memoria RAM de 256 MB instalados, soportando hasta 768 MB.
* Memoria FLASH de 64 MB instalados soportando hasta 256 MB.
* Utiliza IPSec como protocolo de transporte, protección firewall, cifrado del hardware, soporte de MPLS, diseño modular, criptografía de 128 bits, asistencia técnica VPN, filtrado de URL, cifrado de 256 bits y cumple con las normas de la IEEE 802.3af.

**Nota:** Los enrutadores mencionados presentan cambios en el IOS para que soporten MPLS; debido a la antigüedad de los dispositivos, esta medida era necesaria.

# **ANTECEDENTES Y CASO DE ESTUDIO**

La empresa de servicios de internet y telecomunicaciones, TELCOM, solicita un sistema de aprovisionamiento de recursos para una red MPLS, con el objetivo de proveer sus servicios a las empresas clientes. La empresa cliente, CONEX, consta de su matriz en la Urbanización del Salado (URDESA), en la ciudad de Guayaquil; y ha firmado un contrato anual para asistencia a su sucursal recién establecida en el sector Las Piñas en la ciudad de Milagro.

**MPLS:** Es una técnica de transmisión de tráfico IP diseñado para trabajar sobre múltiples protocolos. Destaca su diseño por aspectos como la ingeniería de tráfico, la diferenciación de clases de servicio, y redes virtuales privadas (VPN).

MPLS se sirve de los mejores aspectos de las capas 2 y 3. La principal herramienta que utiliza este protocolo, es la asignación de etiquetas, lo que permite un tráfico fluido y nos servimos de la ingeniería de tráfico para poder decidir, en caso de que se requiera, por dónde enviar el paquete según el etiquetado.

**VLAN:** Es una red de área local, pero de aspecto virtual, es decir que, si bien pertenecen físicamente a una red, lógicamente son independientes según lo amerite la configuración. Nos sirve para segmentar el tráfico de acuerdo a departamentos, por ejemplo.

**VRF:** Las tablas de enrutamiento virtual, son una propiedad de los enrutadores de crear instancias de tabla de enrutamiento independientes de la tabla de enrutamiento global. Su utilidad nos sirve para permitir segmentar el tráfico de manera automática para los diferentes clientes debido a que contiene las rutas pertenecientes a una VPN específica.

## **Diagrama de red**

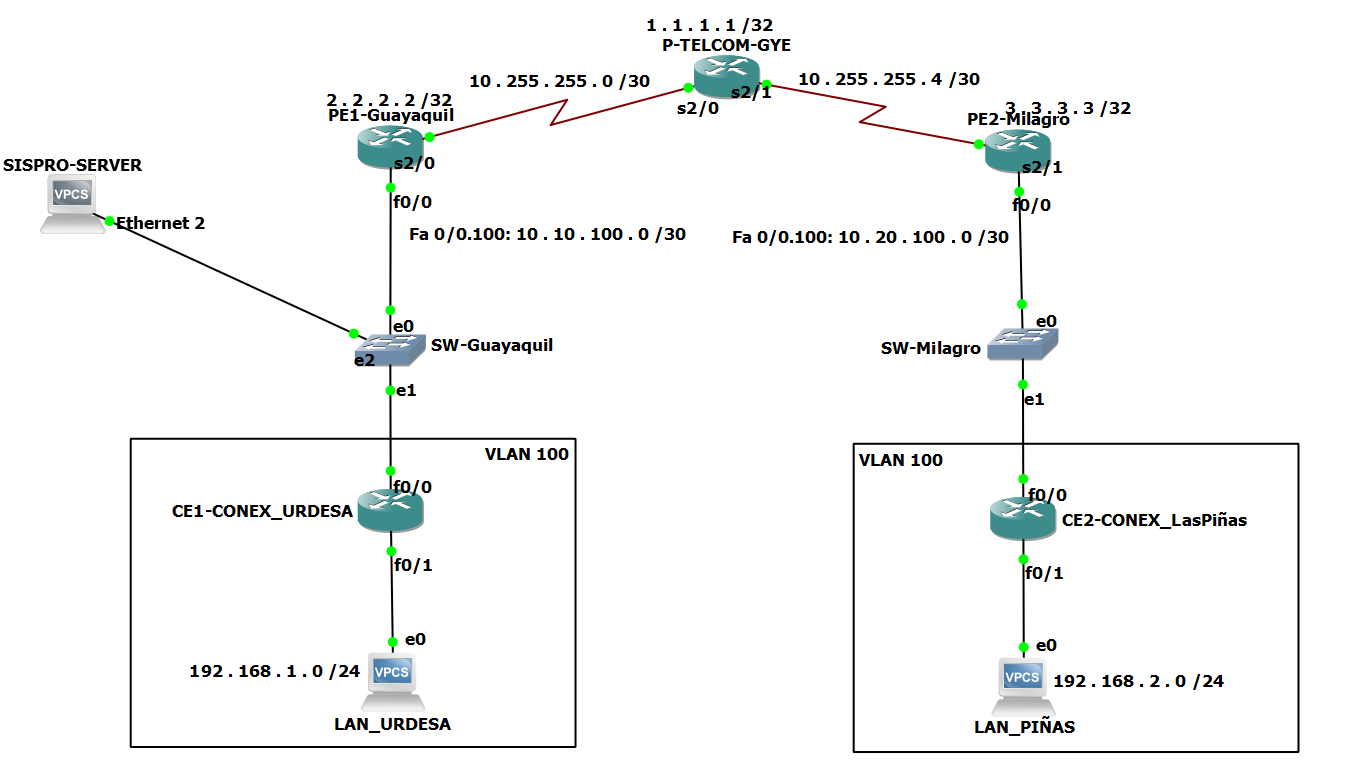


Figura . Diagrama de red

## **Direccionamiento IP**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **DISPOSITIVO** | **INTERFAZ** | **DIRECCIÓN IP** | **MÁSCARA DE SUBRED** | **PUERTA DE ENLACE PREDETERMINADA** |
| **P-TELCOM-GYE** | S0/2/0 | 10.255.255.1 | 255.255.255.252 |  |
| S0/2/1 | 10.255.255.5 | 255.255.255.252 |  |
| Lo0 | 1.1.1.1 | 255.255.255.255 |  |
| **PE1-Guayaquil** | S0/2/0 | 10.255.255.2 | 255.255.255.252 |  |
| Fa0/0.100 | 10.10.100.1 | 255.255.255.252 |  |
| Lo0 | 2.2.2.2 | 255.255.255.255 |  |
| Fa0/0.7 | 10.10.7.1 | 255.255.255.252 |  |
| **PE2-Milagro** | S0/2/1 | 10.255.255.6 | 255.255.255.252 |  |
| Fa0/0.100 | 10.20.100.1 | 255.255.255.252 |  |
| Lo0 | 3.3.3.3 | 255.255.255.255 |  |
| Fa0/0.7 | 10.20.7.1 | 255.255.255.252 |  |
| **CE1-CONEX\_URDESA** | Fa0/0 | 10.10.100.2 | 255.255.255.252 | 10.10.100.1 |
| Fa0/1 | 192.168.1.1 | 255.255.255.0 |  |
| **CE2-CONEX\_LasPiñas** | Fa0/0 | 10.20.100.2 | 255.255.255.252 | 10.20.100.1 |
| Fa0/1 | 192.168.2.1 | 255.255.255.0 |  |

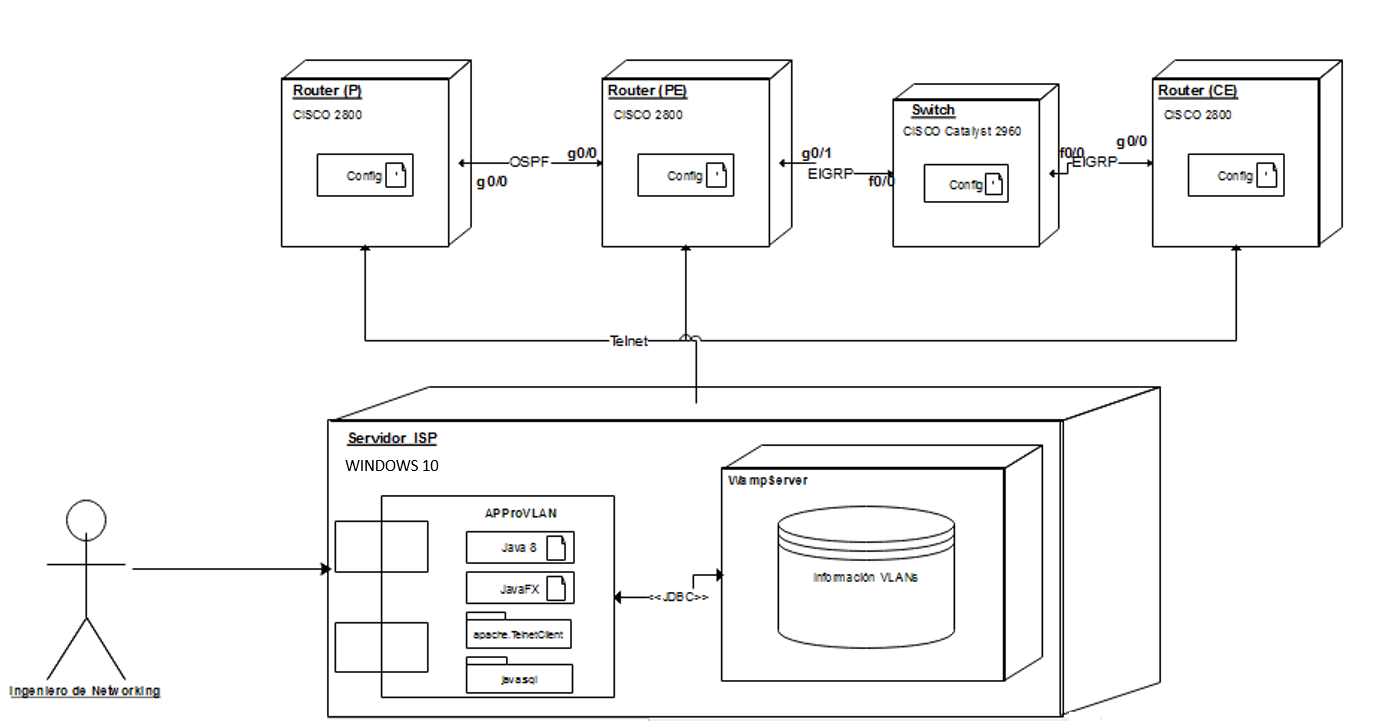
Tabla . Tabla de Direccionamiento IPv4

Bajo la guía que proporcionan el diagrama de red (Figura 1) y la table de direccionamiento (Tabla1), el lector puede replicar el proyecto de una manera controlada, tomando en cuenta que los nombres de las interfaces deben seguir su formato en GNS3 y otro formato para los dipositivos reales y lo que permitan sus marcas. En nuestro caso, la marca de los dispositivos es Cisco, aspecto a tomar en consideración a la hora de configurar interfaces.

Un criterio importante que se aplica es el de la creación de una VLAN para cada empresa; realizándose dicha configuración en los switches respectivos y utilizando subinterfaces para los enrutadores. Para este proyecto se tomaron las siguientes VLAN:

* **VLAN 100:** Red virtual de la empresa CONEX.
* **VLAN 7:** Red virtual del ADMINISTRADOR.

## **Definición de la arquitectura del sistema.**



# **CONFIGURACIONES DE LOS DIPOSITIVOS DE RED**

La sección de este apartado se especifican un orden de configuración que hemos seguido, pero no es obligación seguirlo al pie de la letra.

## **Plantilla Básica**

La siguiente plantilla presenta un conjunto de comandos que asignarán a los conmutadores y enrutadores, una configuración inicial acorde a nuestros requerimientos.

|  |
| --- |
| "DISPOSITIVO"> enable  "DISPOSITIVO"#conf t  "DISPOSITIVO"hostname 'NOMBREDISP'  'NOMBREDISP'(config)# ip name-server 192.168.1.17  'NOMBREDISP'(config)# ip name-server 192.168.1.19  'NOMBREDISP'(config)# ip domain-name "nombre"  'NOMBREDISP'(config)# line console 0  'NOMBREDISP'(config-line)# logging synchronous  'NOMBREDISP'(config-line)# exec-timeout 3 3  'NOMBREDISP'(config-line)# login local |
| **'NOMBREDISP'(config-line)# line vty 0 4**  **'NOMBREDISP'(config-line)# logging synchronous**  **'NOMBREDISP'(config-line)# exec-timeout 3 3**  **'NOMBREDISP'(config-line)# login local**  **'NOMBREDISP'(config-line)# transport input telnet** |
| 'NOMBREDISP'(config-line)# exit  'NOMBREDISP'(config-line)# service password-encryption  'NOMBREDISP'(config-line)# banner motd #ACCESO A PERSONAL AUTORIZADO#  'NOMBREDISP'(config-line)# username admin privilege 15 secret admin  'NOMBREDISP'(config-line)# username monitoreo privilege 5 secret monitoreo |

* **DISPOSITIVO:** Según nuestra plantilla, este campo hace referencia al tipo de dispositivo intermediario que vamos a configurar, esto lo designa el dispositivo físico que usaremos y se verá reflejado en el prompt.
* **NOMBREDISP:** Este campo no tiene que llamarse así, sino que indica que podemos agregarle el nombre que deseemos a los dispositivos.
* La celda de comandos resaltada en negrita, indica que es la configuración necesaria para que el dispositivo acepte protocolos de comunicación como **Telnet,** que usaremos en este proyecto, gracias al comando **transport input telnet**.

## **Configuración de un Provider**

Se detallan las configuraciones para el enrutador Provider (P) de la empresa ubicado en Guayaquil.

**Importante:** A partir de esta sección se mostrarán comandos necesarios para configurar la red solicitada.

* Se ha establecido en las siguientes tablas, lo que denominaremos **celdas de prompt**, una guía para que el lector reconozca en qué modo de ejecutar cada comando.
* En las celdas de comandos, encontrará una sección que dice ‘Aplicativo’, la cual se refiere a que dichos comandos son asignados mediante el programa de aprovisionamiento desarrollado.

**Nota:** En las celdas de prompt muchas veces notarán un asterisco (\*), esto indica que hay que poner atención a la interfaz que se está configurando, ya sea sintaxis o si se cambia de tipo de interfaz o si la marca del dispositivo maneja una propia sintaxis. En este caso, los comandos se aprecian para GNS3 con objetivos de replicar la simulación.

### **Configuración de direccionamiento**

|  |  |
| --- | --- |
| **Nombre de dispositivo:** P-TELCOM-GYE  **Descripción:** Se configura el direccionamiento en las interfaces Loopback y seriales; y en ellas se configura MPLS. | |
| P-TELCOM-GYE(config)#  P-TELCOM-GYE(config-if)#  P-TELCOM-GYE(config-if)# \*  P-TELCOM-GYE(config-if)#  P-TELCOM-GYE(config-if)#  P-TELCOM-GYE(config-if)#  P-TELCOM-GYE(config-if)#  P-TELCOM-GYE(config-if)#  P-TELCOM-GYE(config-if)#  P-TELCOM-GYE(config-if)#  P-TELCOM-GYE(config-if)#  P-TELCOM-GYE(config-if)#  P-TELCOM-GYE(config-if)#  P-TELCOM-GYE(config-if)# | interface Loopback0  ip address 1.1.1.1 255.255.255.255  interface s2/0  description HACIA-PE1-Guayaquil  ip address 10.255.255.1 255.255.255.252  clock rate 64000  no shutdown  mpls ip  interface s2/1  description HACIA-PE2-Milagro  ip address 10.255.255.5 255.255.255.252  clock rate 64000  no shutdown  mpls ip |

### **Configuración de IGP para comunicación entre Provider Edges.**

|  |  |
| --- | --- |
| **Nombre de dispositivo:** P-TELCOM-GYE  **Descripción:** Se anuncian las redes 1.1.1.1/32 y 10.255.255.0/29 por el protocolo de enrutamiento OSPF con AS 1 y en el área de backbone 0. | |
| P-TELCOM-GYE(config)#  P-TELCOM-GYE(config-router)#  P-TELCOM-GYE(config-router)#  P-TELCOM-GYE(config-router)# | router ospf 1  network 1.1.1.1 0.0.0.0 area 0  network 10.255.255.0 0.0.0.7 area 0  exit |

### **Configuración de MPLS y LDP de forma global.**

|  |  |
| --- | --- |
| **Nombre de dispositivo:** P-TELCOM-GYE  **Descripción:** Se configura el protocolo CEF, MPLS y LDP de forma global y por interfaces. | |
| P-TELCOM-GYE(config)#  P-TELCOM-GYE(config)#  P-TELCOM-GYE(config)#  P-TELCOM-GYE(config)# | ip cef  mpls label protocol ldp  mpls ldp router-id Loopback0  mpls ip |

### **Configuración de Customer Edges.**

Se procede a configurar los dispositivos clientes ubicados en URDESA y Las Piñas pertenecientes a Guayaquil y Milagro respectivamente. Estas configuraciones son realizadas manualmente por el personal técnico del ISP pero no con el aplicativo.

|  |  |
| --- | --- |
| **Nombre de dispositivo:** CE1-CONEX\_URDESA  **Descripción:** Se configura el direccionamiento en la interfaz FastEthernet0/0 y FastEthernet0/1, además del protocolo de enrutamiento dinámico EIGRP. | |
| CE1-CONEX\_URDESA (config-if)#  CE1-CONEX\_URDESA (config-if)#  CE1-CONEX\_URDESA (config-if)#  CE1-CONEX\_URDESA (config-if)#  CE1-CONEX\_URDESA (config-if)#  CE1-CONEX\_URDESA (config)#  CE1-CONEX\_URDESA (config-if)#  CE1-CONEX\_URDESA (config-if)#  CE1-CONEX\_URDESA (config-if)#  CE1-CONEX\_URDESA (config-if)#  CE1-CONEX\_URDESA (config)#  CE1-CONEX\_URDESA (config-router)#  CE1-CONEX\_URDESA (config-router)#  CE1-CONEX\_URDESA (config-router)#  CE1-CONEX\_URDESA (config-router)# | interface Fa0/0  description HACIA-SW1  ip address 10.10.100.2 255.255.255.252  no shutdown  exit  interface Fa0/1  description LAN-URDESA  ip address 192.168.1.1 255.255.255.0  no shutdown  exit  router eigrp 10  network 10.10.100.0  network 192.168.1.0  no auto-summary  exit |

|  |  |
| --- | --- |
| **Nombre de dispositivo:** CE1-CONEX\_LasPinas  **Descripción:** Se configura el direccionamiento en la interfaz FastEthernet0/0 y FastEthernet0/1, además del protocolo de enrutamiento dinámico EIGRP. | |
| CE1-CONEX\_ LasPinas(config-if)#  CE1-CONEX\_ LasPinas(config-if)#  CE1-CONEX\_ LasPinas(config-if)#  CE1-CONEX\_ LasPinas(config-if)#  CE1-CONEX\_ LasPinas(config-if)#  CE1-CONEX\_ LasPinas(config)#  CE1-CONEX\_ LasPinas(config-if)#  CE1-CONEX\_ LasPinas(config-if)#  CE1-CONEX\_ LasPinas(config-if)#  CE1-CONEX\_ LasPinas(config-if)#  CE1-CONEX\_ LasPinas(config)#  CE1-CONEX\_ LasPinas(config-router)#  CE1-CONEX\_ LasPinas(config-router)#  CE1-CONEX\_ LasPinas(config-router)#  CE1-CONEX\_ LasPinas(config-router)# | interface Fa0/0  description HACIA-SW1  ip address 10.10.100.2 255.255.255.252  no shutdown  exit  interface Fa0/1  description LAN-URDESA  ip address 192.168.1.1 255.255.255.0  no shutdown  exit  router eigrp 10  network 10.10.100.0  network 192.168.1.0  no auto-summary  exit |

## **Configuración de Provider Edges.**

### **Configuración de direccionamiento para los PE.**

|  |  |  |
| --- | --- | --- |
| **Nombre de dispositivo:** PE1-Guayaquil  **Descripción:** Se configura el direccionamiento de las interfaces Loopback0 y Serial2/0. Adicionalmente se habilita el protocolo MPLS en esta última interfaz porque está conectada a un dispositivo tipo P. | | |
| PE1-Guayaquil(config)#  PE1-Guayaquil(config-if)#  PE1-Guayaquil(config-if)# \*  PE1-Guayaquil(config-if)#  PE1-Guayaquil(config-if)#  PE1-Guayaquil(config-if)#  PE1-Guayaquil(config-if)#  PE1-Guayaquil(config-if)# | | interface Loopback0  ip address 2.2.2.2 255.255.255.255  interface S2/0  description HACIA-P-TELCOM-GYE  ip address 10.255.255.2 255.255.255.252  no shutdown  mpls ip  exit |
| **Nombre de dispositivo:** PE2-Milagro  **Descripción:** Se configura el direccionamiento de las interfaces Loopback0 y Serial2/1. Adicionalmente se habilita el protocolo MPLS en esta última interfaz porque está conectada a un dispositivo tipo P. | | |
| PE2-Milagro(config)#  PE2-Milagro(config-if)#  PE2-Milagro(config-if)#\*  PE2-Milagro(config-if)#  PE2-Milagro(config-if)#  PE2-Milagro(config-if)#  PE2-Milagro(config-if)#  PE2-Milagro(config-if)# | interface Loopback0  ip address 3.3.3.3 255.255.255.255  interface S2/1  description HACIA-P-TELCOM-GYE  ip address 10.255.255.6 255.255.255.252  no shutdown  mpls ip  exit | |

### **Configuración de IGP**

|  |  |  |
| --- | --- | --- |
| **Nombre de dispositivo:** PE1-Guayaquil  **Descripción:** Se establece el protocolo de enrutamiento OSPF con AS 1 y en el área de backbone 0. Se anuncian las redes 2.2.2.2/32 y la 10.255.255.0/30. | | |
| PE1-Guayaquil(config)#  PE1-Guayaquil(config-router)#  PE1-Guayaquil(config-router)#  PE1-Guayaquil(config-router)# | | router ospf 1  network 2.2.2.2 0.0.0.0 area 0  network 10.255.255.0 0.0.0.3 area 0  exit |
| **Nombre de dispositivo:** PE2-Milagro  **Descripción:** Se establece el protocolo de enrutamiento OSPF con AS 1 y en el área de backbone 0. Se anuncian las redes 3.3.3.3/32 y la 10.255.255.4/30. | | |
| PE2-Milagro(config)#  PE2-Milagro(config-router)#  PE2-Milagro(config-router)#  PE2-Milagro(config-router)# | router ospf 1  network 3.3.3.3 0.0.0.0 area 0  network 10.255.255.4 0.0.0.3 area 0  exit | |

### **Comunicación entre Provider Edges.**

|  |  |  |
| --- | --- | --- |
| **Nombre de dispositivo:** PE1-Guayaquil  **Descripción:** Dar a conocer al enrutador los vecinos iBGP con la ip de loopback de cada uno de ellos. En este caso la dirección será 3.3.3.3 según la tabla de direccionamiento. | | |
| PE1-Guayaquil(config)#  PE1-Guayaquil(config-router)#  PE1-Guayaquil(config-router)#  PE1-Guayaquil(config-router)#  PE1-Guayaquil(config-router)#  PE1-Guayaquil(config-router)# | | router bgp 1  neighbor 3.3.3.3 remote-as 1  neighbor 3.3.3.3 update-source Loopback0  neighbor 3.3.3.3 next-hop-self  no auto-summary  exit |
| **Nombre de dispositivo:** PE2-Milagro  **Descripción:** Dar a conocer al enrutador los vecinos iBGP con la ip de loopback de cada uno de ellos. En este caso la dirección será 2.2.2.2 según la tabla de direccionamiento. | | |
| PE2-Milagro(config)#  PE2-Milagro(config-router)#  PE2-Milagro(config-router)#  PE2-Milagro(config-router)#  PE2-Milagro(config-router)#  PE2-Milagro(config-router)# | router bgp 1  neighbor 2.2.2.2 remote-as 1  neighbor 2.2.2.2 update-source Loopback0  neighbor 2.2.2.2 next-hop-self  no auto-summary  exit | |

### **Activación de protocolo de distribución de etiquetas.**

|  |  |  |
| --- | --- | --- |
| **Nombre de dispositivo:** PE1-Guayaquil  **Descripción:** Se configura el protocolo CEF, LDP y MPLS de forma global y por interfaces. | | |
| PE1-Guayaquil(config)#  PE1-Guayaquil(config)#  PE1-Guayaquil(config)#  PE1-Guayaquil(config)# | | ip cef  mpls label protocol ldp  mpls ldp router-id Loopback0  mpls ip |
| **Nombre de dispositivo:** PE2-Milagro  **Descripción:** Se configura el protocolo CEF, LDP y MPLS de forma global y por interfaces. | | |
| PE2-Milagro(config)#  PE2-Milagro(config)#  PE2-Milagro(config)#  PE2-Milagro(config)# | ip cef  mpls label protocol ldp  mpls ldp router-id Loopback0  mpls ip | |

### **Permitir que los enrutadores propaguen información mediante el protocolo MP-BGP.**

|  |  |  |
| --- | --- | --- |
| **Nombre de dispositivo:** PE1-Guayaquil  **Descripción:** Se activa el protocolo MP-BGP para informar sobre rutas y VPNs entre PES. | | |
| PE1-Guayaquil(config)#  PE1-Guayaquil(config-router)#  PE1-Guayaquil(config-router-af)#  PE1-Guayaquil(config-router-af)#  PE1-Guayaquil(config-router-af)#  PE1-Guayaquil(config-router)# | | router bgp 1  address-family vpnv4  neighbor 3.3.3.3 activate  neighbor 3.3.3.3 send-community extended  exit-address-family  exit |
| **Nombre de dispositivo:** PE2-Milagro  **Descripción:** Se activa el protocolo MP-BGP para informar sobre rutas y VPNs entre PES. | | |
| PE2-Milagro(config)#  PE2-Milagro(config-router)#  PE1-Milagro(config-router-af)#  PE1-Milagro(config-router-af)#  PE1-Milagrol(config-router-af)#  PE2-Milagro(config-router)# | router bgp 1  address-family vpnv4  neighbor 2.2.2.2 activate  neighbor 2.2.2.2 send-community extended  exit-address-family  exit | |

### **Creación de tablas de enrutamiento virtuales.**

|  |  |  |
| --- | --- | --- |
| **Nombre de dispositivo:** PE1-Guayaquil  **Descripción:** Se crean las VRF del administrador (ADMINISTRADOR) y de la empresa cliente (CONEX) con router distinguisher para cada uno, y como cada VRF indica exclusividad de tráfico de paquetes, se procede a configurar la subinterfaz correspondiente de esta VRF para cada cliente con su encapsulamiento propio. | | |
| PE1-Guayaquil(config)#  PE1-Guayaquil(config-vrf)#  PE1-Guayaquil(config-vrf)#  PE1-Guayaquil(config-vrf)#  PE1-Guayaquil(config-vrf)#  PE1-Guayaquil(config-vrf)#  PE1-Guayaquil(config)#  PE1-Guayaquil(config-if)#  PE1-Guayaquil(config-if)#  PE1-Guayaquil(config-if)#  PE1-Guayaquil(config-if)#  PE1-Guayaquil(config-if)#  PE1-Guayaqui(config-if)# | | ip vrf ADMINISTRADOR  rd 64000:7  route-target export 7:7  route-target import 7:7  route-target import 100:100  exit  interface f0/0  no shutdown  interface f0/0.7  description PARA-ADMINISTRADOR  encapsulation dot1Q 7  ip vrf forwarding ADMINISTRADOR  ip address 10.10.7.1 255.255.255.252 |
| Aplicativo | | |
| PE1-Guayaquil(config)#  PE1-Guayaquil(config-vrf)#  PE1-Guayaquil(config-vrf)#  PE1-Guayaquil(config-vrf)#  PE1-Guayaquil(config-vrf)#  PE1-Guayaquil(config)#  PE1-Guayaquil(config-if)#  PE1-Guayaquil(config-if)#  PE1-Guayaquil(config-if)#  PE1-Guayaquil(config-if)#  PE1-Guayaquil(config-if)# | | ip vrf CONEX\_100  rd 64000:100  route-target export 100:100  route-target import 100:100  exit  interface f0/0.100  description HACIA-SW1-Guayaquil  encapsulation dot1Q 100  ip vrf forwarding CONEX\_100  ip address 10.10.100.1 255.255.255.252  exit |
| **Nombre de dispositivo:** PE2-Milagro  **Descripción:** Se crean las VRF del administrador (ADMINISTRADOR) y de la empresa cliente (CONEX) con router distinguisher para cada uno, y como cada VRF indica exclusividad de tráfico de paquetes, se procede a configurar la subinterfaz correspondiente de esta VRF para cada cliente con su encapsulamiento propio. | | |
| PE2-Milagro(config)#  PE2-Milagro(config-vrf)#  PE2-Milagro(config-vrf)#  PE2-Milagro(config-vrf)#  PE2-Milagro(config-vrf)#  PE2-Milagro(config-vrf)#  PE2-Milagro(config)#  PE2-Milagro(config-if)#  PE2-Milagro(config-if)#  PE2-Milagro(config-if)#  PE2-Milagro(config-if)#  PE2-Milagro(config-if)#  PE2-Milagro(config-if)# | ip vrf ADMINISTRADOR  rd 64000:7  route-target export 7:7  route-target import 7:7  route-target import 100:100  exit  interface f0/0  no shutdown  interface f0/0.7  description PARA-ADMINISTRADOR  encapsulation dot1Q 7  ip vrf forwarding ADMINISTRADOR  ip address 10.20.7.1 255.255.255.252 | |
| Aplicativo | | |
| PE2-Milagro(config)#  PE2-Milagro(config-vrf)#  PE2-Milagro(config-vrf)#  PE2-Milagro(config-vrf)#  PE2-Milagro(config-vrf)#  PE2-Milagro(config)#  PE2-Milagro(config-if)#  PE2-Milagro(config-if)#  PE2-Milagro(config-if)#  PE2-Milagro(config-if)#  PE2-Milagro(config-if)# | ip vrf CONEX\_100  rd 64000:100  route-target export 100:100  route-target import 100:100  exit  interface f0/0.100  description HACIA-SW1-Guayaquil  encapsulation dot1Q 100  ip vrf forwarding CONEX\_100  ip address 10.20.100.1 255.255.255.252  exit | |

### **Establecer comunicación entre dispositivos PE y CE.**

|  |  |  |
| --- | --- | --- |
| **Nombre de dispositivo:** PE1-Guayaquil  **Descripción:** Se configura como método de enrutamiento dinámico al protocolo EIGRP para la comunicación entre proveedores y clientes. | | |
| PE1-Guayaquil(config)#  PE1-Guayaquil(config-router)#  PE1-Guayaquil(config-router-af)#  PE1-Guayaquil(config-router-af)#  PE1-Guayaquil(config-router-af)#  PE1-Guayaquil(config-router-af)#  PE1-Guayaquil(config-router-af)#  PE1-Guayaquil(config-router-af)#  PE1-Guayaquil(config-router)# | | router eigrp 1  address-family ipv4 vrf ADMINISTRADOR  network 10.10.100.0  no auto-summary  redistribute bgp 1  default-metric 100000 1 255 1 1500  autonomous-system 10  exit-address-family  exit |
| Aplicativo | | |
| PE1-Guayaquil(config)#  PE1-Guayaquil(config-router)#  PE1-Guayaquil(config-router-af)#  PE1-Guayaquil(config-router-af)#  PE1-Guayaquil(config-router-af)#  PE1-Guayaquil(config-router-af)#  PE1-Guayaquil(config-router-af)#  PE1-Guayaquil(config-router-af)#  PE1-Guayaquil(config-router)# | | router eigrp 1  address-family ipv4 vrf CONEX\_100  network 10.10.100.0  no auto-summary  redistribute bgp 1  default-metric 100000 1 255 1 1500  autonomous-system 10  exit-address-family  exit |
| **Nombre de dispositivo:** PE2-Milagro  **Descripción:** Se configura como método de enrutamiento dinámico al protocolo EIGRP para la comunicación entre proveedores y clientes. | | |
| PE2-Milagro(config)#  PE2-Milagro(config-router)#  PE2-Milagro(config-router-af)#  PE2-Milagro(config-router-af)#  PE2-Milagro(config-router-af)#  PE2-Milagro(config-router-af)#  PE2-Milagro(config-router-af)#  PE2-Milagro(config-router-af)#  PE2-Milagro(config-router)# | | router eigrp 1  address-family ipv4 vrf ADMINISTRADOR  network 10.20.100.0  no auto-summary  redistribute bgp 1  default-metric 100000 1 255 1 1500  autonomous-system 10  exit-address-family  exit |
| Aplicativo | | |
| PE2-Milagro(config)#  PE2-Milagro(config-router)#  PE2-Milagro(config-router-af)#  PE2-Milagro(config-router-af)#  PE2-Milagro(config-router-af)#  PE2-Milagro(config-router-af)#  PE2-Milagro(config-router-af)#  PE2-Milagro(config-router-af)#  PE2-Milagro(config-router)# | router eigrp 1  address-family ipv4 vrf CONEX\_100  network 10.20.100.0  no auto-summary  redistribute bgp 1  default-metric 100000 1 255 1 1500  autonomous-system 10  exit-address-family  exit | |

### **Redistribución de rutas con el protocolo MP-BGP**

|  |  |  |
| --- | --- | --- |
| **Nombre de dispositivo:** PE1-Guayaquil  **Descripción**: Las rutas aprendidas del CE a través del proceso EIGRP, serán manejadas con el proceso MP-BGP usando el address-family de las VRFs. | | |
| PE1-Guayaquil(config)#  PE1-Guayaquil(config-router)#  PE1-Guayaquil(config-router-af)#  PE1-Guayaquil(config-router-af)#  PE1-Guayaquil(config-router)# | | router bgp 1  address-family ipv4 vrf ADMINISTRADOR  redistribute eigrp 10  exit-address-family  exit |
| Aplicativo | | |
| PE1-Guayaquil(config)#  PE1-Guayaquil(config-router)#  PE1-Guayaquil(config-router-af)#  PE1-Guayaquil(config-router-af)#  PE1-Guayaquil(config-router)# | | router bgp 1  address-family ipv4 vrf CONEX\_100  redistribute eigrp 10  exit-address-family  exit |
| **Nombre de dispositivo:** PE2-Milagro  **Descripción:** Se configura como método de enrutamiento dinámico al protocolo EIGRP para la comunicación entre proveedores y clientes. | | |
| PE2-Milagro(config)#  PE2-Milagro(config-router)#  PE2-Milagro(config-router-af)#  PE2-Milagro(config-router-af)#  PE2-Milagro(config-router)# | router bgp 1  address-family ipv4 vrf ADMINISTRADOR  redistribute eigrp 10  exit-address-family  exit | |
| Aplicativo | | |
| PE2-Milagro(config)#  PE2-Milagro(config-router)#  PE2-Milagro(config-router-af)#  PE2-Milagro(config-router-af)#  PE2-Milagro(config-router)# | router bgp 1  address-family ipv4 vrf CONEX\_100  redistribute eigrp 10  exit-address-family  exit | |

# **Código Fuente.**

En este apartado se presentan los códigos más importantes para el desarrollo del aprovisionamiento.

|  |
| --- |
| **Clase:** TelnetConnector.java  **Función:** telnetConnecTo()  **Descripción:** Permite realizar la conexión telnet con un ingreso de la dirección IP como parámetro String. |
| **public** **void** telnetConnectTo([String](http://www.google.com/search?hl=en&q=allinurl%3Adocs.oracle.com+javase+docs+api+string) ip){         **try** {             /\*PERMITE REALIZAR LA CONECION TELNET\*/             telnet.connect(ip);              in = telnet.getInputStream();             out = **new** [PrintStream](http://www.google.com/search?hl=en&q=allinurl%3Adocs.oracle.com+javase+docs+api+printstream)(telnet.getOutputStream());             **this**.state\_telnet\_con=**true**;                      } **catch** ([IOException](http://www.google.com/search?hl=en&q=allinurl%3Adocs.oracle.com+javase+docs+api+ioexception) ex) {             **this**.state\_telnet\_con=**false**;         }     } |
| **Clase:** TelnetConnector.java  **Función:** sendCommand()  **Descripción:** Permite enviar el comando que programamos en el aplicativo hacia el enrutador que esté conectado vía Telnet. |
| public void sendCommand(String prompt\_,String command\_or\_data, Boolean auth) {    try{  System.out.print(readUntil(prompt\_));  write(command\_or\_data);  }catch(Exception e){  System.out.println("Error al enviar el comando");  } |
| **Clase:** TelnetConnector.java  **Función:** authUser()  **Descripción:** Permite la autenticación de las credenciales establecidas para la conexión remota. |
| public void authUser(String user, String pass){    /\*SE ENVIAN LOS COMANDOS RESPECTIVOS AL DISPOSITIVO\*/  this.sendCommand("Username:", user,true);  this.sendCommand("Password:", pass, true);    } |

|  |
| --- |
|  |
|  |