

Simulador de dominio MPLS

MPLS domain simulator

MENDOZA-AUSTRIA, Luis*†, PÉREZ-HERNÁNDEZ, Darío, LEÓN-OLIVARES, Eric y MARTINEZ-PAGOLA, Salvador

Instituto Tecnológico de Pachuca

ID 1^{er} Autor: *Luis, Mendoza-Austria* / ORC ID: 0000-0002-1572-2159, Researcher ID Thomson: P-4842-2018, NCBI: mendozaaustria

ID 2^{do} Coautor: *Darío, Perez-Hernández* / ORC ID: 0000-0003-2147-3141, Researcher ID Thomson: P-6763-2018, NCBI: darioph

ID 3^{er} Coautor: *Eric, León-Olivares* / ORC ID: 0000-0003-4937-0996, Researcher ID Thomson: 6239-2018, arXiv: smpaola2000#1

ID 4^o Coautor: *Salvador, Martinez-Pagola* / ORC ID: 0000-0002-6342-3719, Researcher ID Thomson: L-7445-2018, MXINF2704

Recibido: 01 de Octubre, 2018; Aceptado 02 de Diciembre, 2018

Resumen

La conmutación de etiquetas multiprotocolo (MPLS) es un mecanismo de transporte de datos estándar creado por la IETF, con la finalidad de dar solución a problemas de conmutación. En este proyecto se ha desarrollado un software de Simulación de Dominios MPLS, encargado de actuar como una Red MPLS virtual, con las funciones básicas de esta tecnología de tal manera que el estudiante comprenda la función básica de la misma y simultáneamente sirva como una herramienta didáctica en la enseñanza-aprendizaje para el docente que imparte la materia de redes. Para la comprensión del trabajo que se realizó dentro de un Dominio MPLS, se muestra una introducción a tema y un marco teórico de MPLS, el análisis del algoritmo de multietiquetado, el diseño del software, la programación básica del simulador, y una muestra de los resultados.

Dominio, MPLS, IETF, Simulador, Virtual, Didáctica

Abstract

Multiprotocol label switching (MPLS) is a standard data transport mechanism created by the IETF, with the purpose of solving switching problems. In this project a software of Simulation of Domains MPLS has been developed, in charge of acting as a virtual MPLS Network, with the basic functions of this technology in such a way that the student understands the basic function of the same and simultaneously serves as a didactic tool in teaching-learning for the teacher who teaches the subject of networks. For the understanding of the work that was done within an MPLS Domain, an introduction to the subject and a theoretical MPLS framework, the analysis of the multi-labeling algorithm, the software design, the basic simulator programming, and a sample of the results.

Domain, MPLS, IETF, Simulator, Virtual, Didactic

Citación: MENDOZA-AUSTRIA, Luis, PÉREZ-HERNÁNDEZ, Darío, LEÓN-OLIVARES, Eric y MARTINEZ-PAGOLA, Salvador. Simulador de dominio MPLS. Revista de Simulación Computacional. 2018. 2-6: 25-31

* Correspondencia al autor (correo electrónico: mendozaaustrial@hotmail.com)

† Investigador contribuyendo como primer autor.

Introducción

La conmutación de etiquetas multiprotocolo (MPLS) es un mecanismo de transporte de datos estándar creado por la IETF, con la finalidad de dar solución a problemas de conmutación.

Para la comprensión del trabajo que se realiza dentro de un Dominio MPLS, existen videos e información documentada a cerca de la funcionalidad de esta tecnología, que son las bases con las que el Docente trabaja a fin de explicar el objetivo de un dominio MPLS, sin embargo, en muchas ocasiones es insuficiente, puesto que el alumno requiere del dinamismo y de la interactividad con lo que estudia, para tales efectos se le da lugar a un Simulador de Dominios MPLS, encargado de actuar como una Red MPLS virtual, con las funciones básicas de esta tecnología de tal manera que el alumno comprenda la función básica de la misma y simultáneamente sirva como una herramienta didáctica para el docente.

MPLS (Conmutación de etiquetas multiprotocolo)

MPLS es una tecnología que modifica el reenvío tradicional de paquetes que analiza la dirección IP de destino contenida en la cabecera de Capa de Red de cada paquete, por medio de la cual un paquete viaja desde la fuente hasta su destino final. Los protocolos de enrutamiento dinámicos o estáticos construyen una base de datos necesaria la cual se analiza para tomar una decisión hacia dónde va el paquete IP según la dirección de destino, dicha tabla se conoce como tabla de enrutamiento.

En el reenvío de paquetes usado por MPLS se puede apreciar una diferencia drástica con el reenvío original de paquetes dado que en este último entorno cada paquete es analizado en cada uno de los saltos de la red donde se checa la cabecera Capa 3, y con base en la información de la misma se toma una decisión conforme a la tabla de enrutamiento de cada dispositivo de Capa 3.

Trabaja entre la capa de enlace de datos y la capa de red del modelo OSI.



Figura 1 Modelo Osi con MPLS

Mecanismo de imposición de etiquetas en MPLS

La imposición de etiquetas en MPLS es la acción de añadir una etiqueta a un paquete cuando éste entra a un dominio MPLS. Esta función se realiza en los límites de la red MPLS y es ejecutada por un Edge-LSR.

En el reenvío tradicional de paquetes IP, cada salto en la red realiza una consulta en la tabla de reenvío IP, y con base en la dirección destino que se encuentra en la cabecera de red, selecciona el siguiente salto y reenvía el paquete hacia éste. Esta iteración se repite en cada salto de la red hasta que el paquete llega a su destino final.

Escoger el siguiente salto para el paquete IP es la combinación de dos funciones, la primera función clasifica todos los paquetes que llegan en determinado momento al router en varios grupos de prefijos IP destino; es decir, agrupa todos los paquetes que pertenecen a una misma subred y que por lo tanto tienen que ser reenviados al mismo destino, la segunda función asocia a cada grupo de paquetes creado en el primer paso a una dirección IP que es su siguiente salto.

Dentro de la arquitectura MPLS, el resultado de la primera función, es decir, los grupos de paquetes con el mismo destino es llamado un FEC (Forwarding Equivalent Classes o clase equivalente de reenvío), cada paquete dentro de un FEC es reenviado de la misma manera, por lo tanto, atraviesan la Red usando la misma ruta.

A diferencia del reenvío tradicional de paquetes, el proceso de asignar a un paquete un FEC es realizado sólo una vez y no por cada salto, con lo cual se reduce el procesamiento de cada router dentro de la red. El FEC al cual el paquete es asignado es luego codificado como un identificador de tamaño fijo llamado etiqueta.

Cuando el paquete llega al siguiente salto dentro del dominio MPLS, el LSR que lo recibe analiza su etiqueta y lo reenvía al siguiente salto dependiendo de la información que aparece en la LIB (Label Information Base o Base de información de etiquetas). Es muy importante anotar, que este análisis se realiza primero que el de Capa 3, por lo tanto, el proceso de toma de decisión a nivel 3 no se realiza.

Los paquetes MPLS entran en la red a través de un LSR de entrada y salen de ella a través de un LSR de salida. El camino que toma un paquete de un lado a otro se denomina LSP (Label Switched Path) ó ruta de conmutación de etiquetas, éste path es construido a partir de la información que se toma de una FEC.

Un LSP trabaja en un esquema orientado a conexión, es decir que el path tiene que ser formado antes de que cualquier flujo de tráfico empiece a circular por éste. Cuando un paquete atraviesa la red MPLS, cada LSR cambia la etiqueta entrante por una nueva etiqueta saliente, este proceso continúa hasta que el último LSR ha sido alcanzado.

Cada LSR mantiene dos tablas que soportan toda la información relevante al componente de reenvío MPLS. La primera conocida como LIB (Label Information Base) ó Base de información de etiquetas, donde se guardan todas las etiquetas asignadas por este LSR y las correspondencias de esas etiquetas a otras recibidas desde algún LSR vecino. Las correspondencias de estas etiquetas son distribuidas por medio de protocolos para este uso específico.

La segunda tabla conocida como LFIB (Label Forwarding Information Base) ó Base de información de reenvío de etiquetas y en la cual son mantenidas únicamente las etiquetas que están siendo actualmente usadas por el componente de reenvío. Las LFIB son el equivalente MPLS de una matriz de conmutación en un Switch ATM.

Componentes de una red MPLS

Se le llaman enrutadores de conmutación de etiquetas o LSR (Label Switch Router) a cualquier dispositivo que esté involucrado en el proceso de distribución de etiquetas y que pueda reenviar paquetes. La función básica del proceso de distribución de etiquetas es poder permitir a cada LSR que distribuya sus vínculos de etiquetas a otros LSRs dentro de la red MPLS.

Existen varios tipos de LSRs que se diferencian por las funciones que cada uno de ellos desempeña, entre ellos están el Edge-LSR, ATM-LSR y ATM Edge-LSR, la diferencia entre ellos es puramente a nivel de arquitectura, es decir el mismo dispositivo puede ejercer cualquiera de los roles descritos.

El Edge-LSR es un router que realiza la imposición o la remoción de las etiquetas en los límites de la red MPLS, la labor de imposición consiste en añadir al paquete original una etiqueta o un conjunto de etiquetas en el punto de ingreso a la red. La función de extracción es lo contrario y consiste en quitar la última etiqueta del paquete en el punto de egreso antes de ser reenviada al siguiente host externo a la red MPLS.

Cualquier LSR que esté contiguo a un nodo no MPLS es considerado un Edge-LSR. Sin embargo, si ese LSR tiene alguna interfaz conectada a un ATM-LSR, entonces toma el nombre de ATM Edge-LSR. Los Edge-LSR usan la tabla de reenvío IP tradicional para etiquetar paquetes IP o para remover las etiquetas de los mismos antes de ser enviados a un nodo no MPLS.

Un ATM-LSR es un switch ATM que puede actuar como un LSR, es decir, es un dispositivo que realiza enrutamiento IP y asignación de etiquetas en el componente de control, y reenvío de paquetes de datos usando los mecanismos de conmutación de paquetes propios de ATM, para ésto usa su matriz de conmutación ATM como su LFT (Tabla de reenvío de etiquetas).

Dominio MPLS

Conjunto de Nodos con funcionalidad MPLS y que pertenecen a un mismo Dominio de Encaminamiento IP, nodos adyacentes.

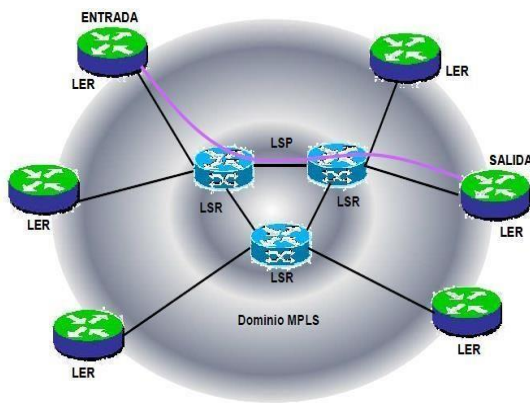


Figura 2 componentes del dominio MPLS

Para MPLS en modo trama, las etiquetas se insertan antes de la cabecera IP, mientras que para ATM es en modo Celda, donde la etiqueta es la misma dirección VPI/VCI y para Frame Relay la etiqueta es el DLCI. En la siguiente figura se representa el esquema de los campos de la cabecera genérica MPLS y su relación con las cabeceras de los otros niveles.

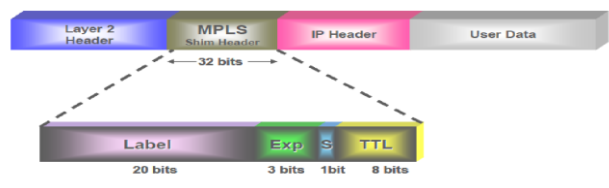


Figura 3 Etiqueta MPLS

API JAVA 2D

API Java 2D es un conjunto de clases para gráficos e imágenes 2D avanzados. Abarca arte lineal, texto e imágenes en un solo modelo integral. La API proporciona una amplia compatibilidad con la composición de imágenes y las imágenes de canales alfa (define la opacidad de un pixel en la imagen), para programas Java a través de extensiones de Abstract Windowing Toolkit (AWT Kit de herramientas de ventana abstracta), un conjunto de clases para proporcionar una definición y conversión precisa del espacio de color y un amplio conjunto de operadores de generación de imágenes orientados a la visualización.

Los objetos 2D de Java existen en un plano llamado espacio de coordenadas del usuario, o simplemente espacio de usuario. Cuando los objetos se representan en una pantalla o impresora, las coordenadas de espacio de usuario se transforman en coordenadas de espacio del dispositivo.

La API Java 2D también permite la creación de bibliotecas gráficas avanzadas, como bibliotecas CAD-CAM (Computer-Aided Design, Computer-Aided Manufacturing) y bibliotecas de efectos especiales de imágenes o gráficos, así como la creación de filtros de lectura/ escritura de archivos gráficos y de imágenes.

Los paquetes de API DE JAVA 2D

Las clases de Java 2D API están organizadas en los siguientes paquetes:

- java.awt
- java.awt.geom
- java.awt.font
- java.awt.color
- java.awt.image
- java.awt.image.renderable
- java.awt.print

El paquete java.awt contiene las clases e interfaces API Java 2D que son de naturaleza general o que mejoran las clases heredadas. (Obviamente, no todas las clases en java.awt son clases Java 2D).

El paquete java.awt.geom contiene clases e interfaces relacionadas con la definición de primitivas geométricas.

El paquete java.awt.font contiene clases e interfaces utilizadas para el diseño de texto y la definición de fuentes.

El paquete java.awt.color contiene clases e interfaces para la definición de espacios de color y perfiles de color.

Los paquetes java.awt.image y java.awt.image.renderable contienen clases e interfaces para la definición y representación de imágenes.

El paquete java.awt.print contiene clases e interfaces que permiten imprimir todo el texto, gráficos e imágenes basados en 2D de Java.

Programación

La idea de este proyecto se originó bajo la necesidad de una herramienta didáctica para la enseñanza del tema de los dominios MPLS, ya que siendo un tema completamente nuevo para los estudiantes resulta de cierto modo difícil su comprensión por la gran cantidad de conceptos nuevos que esta tecnología conlleva.

En el desarrollo de este software se hizo uso de los conocimientos que en la carrera (Ingeniería en Sistemas Computacionales) se han adquirido como lo es: el uso de variables estáticas, herencia, trazo de líneas y desplazamiento de objetos dentro de un panel, algoritmos de búsqueda, matrices de adyacencia, etc.

El software fue desarrollado en lenguaje Java a través del IDE NetBeans y es importante destacar que carece de algunas funciones que se aun pretenden implementar.

A continuación se presenta una parte del código generado en donde se aplica el algoritmo DFS (Depth First Search) o búsqueda en profundidad que es ocupado para establecer la ruta de los FEC desde su origen hasta su destino, una vez encontrada la ruta a seguir se pinta mediante otros procedimientos.

CODIGO DE CLASE: SimulatorMPLS.java

```
//Función que retorna la ruta para el FEC
public ArrayList<String> buscaRuta (String origen,
String destino){
//Variables empleadas en el algoritmo
String nodo;
ArrayList<String> ruta = new ArrayList();
ArrayList<String> vecinos = new ArrayList();
//Pila empleada en el procedimiento
Stack s = new Stack (); ruta.add(origen);
s.push(ruta);

//Mientras el último nodo de la ruta no sea el destino
while(!ruta.get(ruta.size()-1).equals(destino)){ //Se
obtiene elemento de la pila ruta=(ArrayList<String>)
s.pop(); //tomamos el último nodo
nodo=ruta.get(ruta.size()-1);
//obtenemos sus nodos vecinos
vecinos = buscaVecinos(nodo, ruta);
//recorre los nodos vecinos encontrados
for (int j = 0; j < vecinos.size(); j++) {
//Si encontramos al destino
if(vecinos.get(j).equals(destino)){
//Lo agregamos a la ruta
ruta.add(vecinos.get(j));
break; //Salimos del ciclo }
else{//Si no, agregamos el nodo a la ruta y esta como
nuevo elemento a la pila
ArrayList<String> ra = new ArrayList();
ra=ruta; ra.add(vecinos.get(j)); s.push(ra); }}}

//Cuando se encuentre la ruta se devuelve return ruta; }
//Función que devuelve los nodos vecinos de un nodo
private ArrayList<String> buscaVecinos (String nodo,
ArrayList<String> ruta) {
boolean pertenece=false;
ArrayList<String> vecinos = new ArrayList();
```

```
//recorre lista origen (enlaces entre routers)
//de la clase Component definida
for (int j = 0; j < Component.origen.size(); j++) {
if(Component.origen.get(j).equals(nodo)){
for(int x=0; x<ruta.size();x++){
//Busca los nodos adyacentes sin que estén incluidos
en la lista "ruta" que es la ruta actual.
if(Component.destino.get(j).equals(ruta.get(x))){
pertenece=true;
break;}}
if(!pertenece){
//Se agregan a la lista que se retornara
vecinos.add(Component.destino.get(j)); }}} return
vecinos;}
```

Resultados

Finalmente se presentan las imágenes del simulador desarrollado, en ellas se puede apreciar la creación de la Conmutación de etiquetas multiprotocolo:

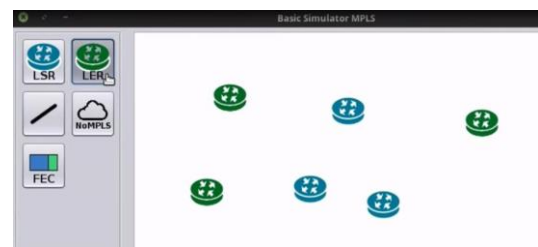


Figura 4 Etiqueta MPLS



Agregando redes no MPLS

Figura 5 Agregando redes no MPLS



Añidiendo paquetes al FEC

Figura 6 Agregando paquetes al FEC

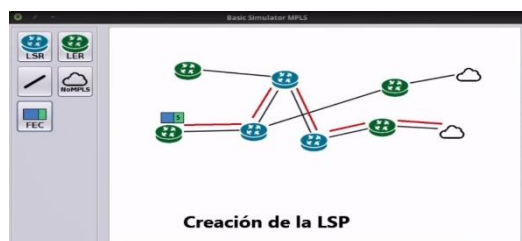


Figura 7 creación de la LSP

Para la obtención de resultados y para conocer el éxito del proyecto se llevó a cabo una pequeña encuesta a algunos alumnos que anteriormente tomaron la materia de Redes Emergentes y VoIP, materia en la cual uno de los primeros temas vistos es el uso de redes MPLS.

1. ¿Cómo considera el diseño del simulador de Dominios MPLS?

a) Excelente b) Bueno c) Regular d) Malo

2. ¿Considera que el uso de esta herramienta puede ayudar en el aprendizaje de los alumnos que cursen materias relacionadas al tema en cuestión?

a) Si b) No ¿Por qué?

3. ¿Le gustaría que los profesores utilizaran este tipo de simuladores, no solamente para las redes MPLS, sino para otro tipo de redes vistas en clase?

a) Si b) No ¿Por qué?

4. ¿Cree usted que el simulador cumple con las características básicas de un Dominio MPLS?

a) Si b) No

5. ¿Qué considera que necesita mejorar el simulador?

XVI. ANÁLISIS DE RESULTADOS

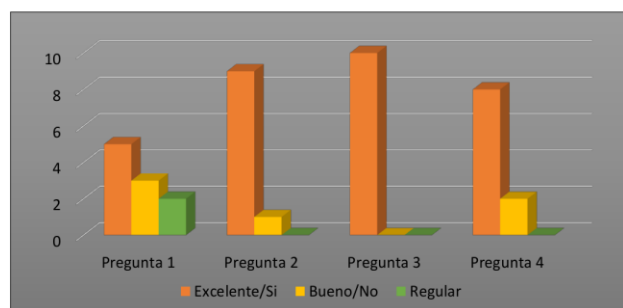


Grafico 1 Análisis de resultados

Una vez aplicadas las encuestas a los alumnos que interactuaron con el sistema pudimos observar los siguientes resultados:

En la pregunta número 1 respecto al diseño del simulador, la gran mayoría respondió que era excelente, sin embargo, es la pregunta con mayor índice de respuestas “regular”, por lo cual es necesaria una mejora en ese campo.

En la pregunta número 2 obtuvimos los resultados esperados, pues ésta se relaciona en gran medida con el objetivo general, el cual es ayudar a través del Simulador MPLS al aprendizaje de los alumnos.

La gran mayoría coincidió en que el uso de este tipo de herramientas didácticas (simuladores) sería de gran provecho, aplicadas a diferentes materias y con los parámetros necesarios, pues resulta más sencillo aprender al interactuar con lo que se está estudiado.

Obtuvimos resultados favorables en la pregunta número 4, sin embargo, es necesario poner atención especial en dicho punto a que también existieron respuestas desfavorables.

Conclusiones

Después de analizar los resultados obtenidos, podemos afirmar que el Simulador MPLS fue de gran utilidad en la comprensión del tema de Redes MPLS, sin embargo, recibimos distintos aportes después de aplicar las encuestas a aquellos que habían interactuado con el Simulador de dominios, los cuales sirvieron para concretar nuevas ideas y realizar cambios que mejoraran el proyecto.

La creación del Simulador definitivamente obtuvo buenos resultados al ser la sustitución de una red MPLS física de alto costo, pues éste se basa únicamente en la manipulación de un software que ofrece las posibilidades básicas de una red MPLS.

Pese a que el simulador tuvo como prueba piloto a una pequeña cantidad de alumnos de la carrera de Ingeniería en Sistemas Computacionales, estos comprendieron los conceptos básicos, haciendo hincapié en el hecho de que el Simulador está pensado como una herramienta didáctica, es decir, requiere la intervención parcial del docente para que el tema quede cien por ciento entendible.

Finalmente cabe destacar que el Simulador MPLS se encuentra en una etapa de mejora, pues, aunque cumple con las expectativas iniciales, es posible mejorarlo para un mejor aprovechamiento.

Referencias

García Tomas Jesús, Raya Cabrera José Luís, Raya Víctor Rodrigo. (2002). Referenciado de Alta Velocidad y Calidad de Servicio en Redes IP, Alfa omega,

<http://cursos.itam.mx/licenciatura/uciel/material/Switches/swl3.pdf>

<http://cursos.itam.mx/licenciatura/uciel/material/Switches/swl3.pdf>

<http://ietf.org/html.chapters/mpls-charter.html>
<http://ManoloDominguez.com>

http://www.univalle.edu.co/~telecomunicaciones/tg_FernandoArevalo.pdf - pag164

Leonardo Jiménez Orozco. (Diciembre de 2000). .MPLS: Una Nueva Tecnología Aplicada a Internet 2. www.danysoft.com

Manolo Domínguez Dorado. (España, 2004) Soporte de Garantía de Servicio (GoS) sobre MPLS mediante Técnicas Activas.

Maria Sol Canalis. MPLS “Multiprotocol Label Switching”: Una Arquitectura de Backbone para la Internet del Siglo XXI.. Dpto Informatica Universidad Nacional del Nordeste. Argentina.
9

Orozco Jiménez, Leonardo. (Diciembre de 2000). MPLS: Una nueva tecnología aplicada a Internet 2. Tesis de Grado, Universidad Autónoma de Baja California., paginas 18-19.