

Adaptaciones del Protocolo BGP-4 para Reducir la Congestión en Redes IP



**Andrés Arturo Delgado Vallejo
Daniel Andrés Sánchez Sánchez**

Universidad del Cauca
Facultad de Ingeniería Electrónica y Telecomunicaciones
Departamento de Telecomunicaciones
Grupo de Nuevas Tecnologías en Telecomunicaciones
Popayán, Enero 2010

Adaptaciones del Protocolo BGP-4 para Reducir la Congestión en Redes IP

**Andrés Arturo Delgado Vallejo
Daniel Andrés Sánchez Sánchez**

**ANEXO A
INTRODUCCIÓN A LAS HERRAMIENTAS DE SIMULACIÓN,
SELECCIÓN DE LA HERRAMIENTA Y CÓDIGOS DML**

Director: Ing. Jenny Cuatindioy Imbachi

Universidad del Cauca
Facultad de Ingeniería Electrónica y Telecomunicaciones
Departamento de Telecomunicaciones
Grupo de Nuevas Tecnologías en Telecomunicaciones

Popayán, Enero 2010

Tabla De Contenido

1. HERRAMIENTAS DE SIMULACIÓN	2
1.1. INTRODUCCIÓN A LAS HERRAMIENTAS DE SIMULACIÓN.....	2
1.2. HERRAMIENTAS DE SIMULACIÓN	2
2. HERRAMIENTA DE SIMULACIÓN SELECCIONADA Y SUS CARACTERÍSTICAS.	3
2.1. ARQUITECTURA Y CAPAS DEL SIMULADOR DE SSFNET.	4
2.2. DIRECCIONAMIENTO NHI EN SSFNET.....	5
3. CONFIGURACIÓN Y EJECUCIÓN DE UNA SIMULACIÓN EN SSFNET	6
3.1. CREACIÓN DEL MODELO DE RED.....	6
3.2. SSF RACEWAY	6
3.3. EL COMPONENTE BGP	7
3.4. KIT DESARROLLADOR DE JAVA.....	7
3.5. CONFIGURACIÓN Y EJECUCIÓN DEL SIMULADOR.	8
3.6. RESULTADOS DEL SIMULADOR.....	9
4. ESCENARIOS IMPLEMENTADOS, CÓDIGOS DML.	11
4.1. ESCENARIO 1	11
4.2. ESCENARIO 2	30
4.3. ESCENARIO 3	41
4.4. ESCENARIO 4	54
BIBLIOGRAFIA.....	67

Índice De Figuras

Figura 1 Capas de Simulación de SSFNet	5
Figura 2 Enlace NHI entre dos enrutadores.....	6
Figura 3 Número de paquetes entregados.....	10
Figura 4 Tablas de enrutamiento de equipos BGP.....	10
Figura 5 Escenario de simulación 1.	11
Figura 6 Escenario de simulación 2.	30
Figura 7 Escenario de simulación 3	41
Figura 8 Escenario de simulación 4	54

ANEXO A

INTRODUCCIÓN A LAS HERRAMIENTAS DE SIMULACIÓN, SELECCIÓN DE LA HERRAMIENTA Y CÓDIGOS DML

1. HERRAMIENTAS DE SIMULACIÓN

1.1. INTRODUCCIÓN A LAS HERRAMIENTAS DE SIMULACIÓN

En un proceso de investigación formal en el área de redes de telecomunicaciones, se hace obligatorio la ejecución de un escenario que reúna las características del estudio realizado, poniendo a prueba la red junto con las modificaciones pertinentes con el objetivo de obtener los resultados que el investigador en algún momento espera y prever situaciones que pueden comprometer el desempeño de la red.

Par tal efecto, los programas de simulación en estas áreas brindan un servicio esencial en cuanto a estudios y pruebas que se realicen sobre una red, con características específicas de QoS, tratamiento diferencial de tráfico, procesos de enrutamiento, entre otros.

Y este en este proyecto no hay excepción, se hace necesaria una herramienta de simulación capaz de soportar las adaptaciones protocolo BGP-4, diferentes escenarios de red con múltiples dominios interactuando entre sí y con el tráfico de datos tan voluminoso como para obtener resultados coherentes en la búsqueda de la reducción de la congestión en las redes IP, y proceder a evaluarlos para respaldar los estudios de esta investigación.

A continuación se describen brevemente las herramientas software de simulación tenidas en cuenta para este proyecto, sus características y la herramienta seleccionada finalmente.

1.2. HERRAMIENTAS DE SIMULACIÓN

Si bien en Internet se pueden encontrar herramientas de simulación para este tipo de proyectos, software tan robusto capaz de recrear escenarios de telecomunicaciones, evaluar comportamientos, analizar procesos y entregar resultados de diferentes formas; se debe tener en claro que esta clase de herramientas son licenciadas y que su costo es bastante elevado para un proyecto de investigación como el presente.

Las empresas que crean y venden este software, tienen versiones especiales dirigidas a la comunidad académica y científica, sin embargo, resultan muy limitadas en algunos casos y con razón; las empresas proveen estas herramientas con un costo muy bajo o en algunos casos las versiones especiales tienen un periodo de prueba de algunos días, pero esto no significa que sean gratis, por tanto no se pueden disponer de ellas cuando se desee.

Se han analizado varias herramientas de simulación y se describen brevemente sus aspectos más relevantes.

- **OPNET IT Guru AE:** el software de simulación de la empresa OPNET tiene para el sector académico el IT Guru Edición Académica (AE: Academic Edition), entre sus características más importantes se encuentran que trae consigo una extensa librería de dispositivos de red, permite al usuario modelar distintos escenarios de red y realizar

diferentes pruebas bajo las condiciones de carga que se especifiquen, además de la interfaz de usuario sencilla y entendible por medio del editor de proyectos; sin embargo, el problema principal de este software es que se encuentra muy limitado en funcionalidades y los cambios que se necesiten realizar al código de funcionamiento, en este caso, no permite cambios al algoritmo del protocolo BGP por ser ésta la versión libre del producto [1].

- **NS-2:** Este simulador de red desarrollado en código C++, para plataforma Windows, Linux y Solaris, es uno de los más completos y flexibles programas de este tipo, considerando una de las características más importantes es el manejo de scripts tipo OTcl línea por línea, representando los enlaces entre los nodos con el manejo de aspectos tan importantes como retardos, pérdidas, gestión de colas, errores, entre otros; cabe aclarar que si se necesita realizar modificaciones a uno de los atributos del objeto OTcl, debe ser por medio de código que maneja el programa, es decir C++. El Planificador de Eventos (Event Scheduler) es una parte esencial de este software, debido a que es el encargado de ejecutar todos los eventos programados en el script OTcl en un determinado periodo de simulación, [57]; esta herramienta implementa provee soporte para protocolos de enrutamiento como BGP, OSPF, RIP y protocolos de transporte como UDP y TCP, fuentes de tráfico como FTP, Telnet, Web, CBR y VBR, además de las diversas políticas de gestión de colas. Los resultados pueden ser visualizados de manera gráfica y animada ya sea por programas como el Animador de Red (NAM: Network Animator) o el XGRAF [2] – [3].
- **SSFNet:** el marco de simulación escalable (SSFNet: Scalable Simulation Framework), es uno de los programas de simulación más completos e importantes actualmente, permitiendo modelar y simular grandes redes de una manera muy completa, basado en JAVA y con un alto grado de granularidad a nivel de paquetes IP, una gran escalabilidad y permitiendo una configuración de atributos de los componentes de la red, entre otras importantes características que hacen de este simulador una herramienta robusta en esta área. Siendo un simulador de eventos discretos, está integrado por dos interfaces, la primera son los API's de programación en lenguajes JAVA y C++, y la segunda es el lenguaje de modelado llamado DML (DML: Domain Modeling Language), en el cual están hechos los archivos de configuración de red, y usados para configurar un modelo completo de un escenario de telecomunicación, e iniciar una simulación con la ayuda de la base de datos escalable DML que se encuentra incluida dentro del simulador SSFNet [4].

2. HERRAMIENTA DE SIMULACIÓN SELECCIONADA Y SUS CARACTERÍSTICAS.

Gracias a estudios previos al presente trabajo de grado y la información del mismo software, se lograron identificar ciertas características esenciales para seleccionar la herramienta de simulación adecuada, en búsqueda de respaldar las modificaciones al atributo BGP propuestas en este documento.

La herramienta seleccionada fue SSFNet, la cual fue utilizada en anteriores investigaciones permitiendo realizar una simulación con datos conocidos, elementos comunes, evaluar los resultados obtenidos y posteriormente realizar una comparación con las modificaciones al protocolo BGP-4, propuestas en anteriores estudios con diferentes escenarios, lo cual permitió seleccionar a SSFNet como el simulador adecuado para esta investigación.

Un punto de cierta importancia que implicaba la selección de la herramienta, eran los requerimientos de instalación para el computador, se debe tener instalado Windows XP Service Pack 2 junto con una herramienta llamada Kit Desarrollador de Java en su versión 1.2 (JDK: Java Development Kit), además de características de espacio en el disco y velocidad con las que se contaban en el momento.

A continuación serán descritas las características más relevantes de este simulador, las cuales son de gran apoyo en el momento de evaluar la modificación al protocolo BGP-4 consignada en este trabajo de grado.

Como se mencionaba anteriormente, el software SSFNet, es un programa que gracias a su estabilidad, escalabilidad y compatibilidad con la mayoría de los protocolos de comunicación, se ha posicionado como uno de los mejores programas de simulación; trabajando con API's de simulación en lenguajes JAVA y C++ y el lenguaje de modelado DML [4].

Se emplean dos clases principalmente para construir cualquier modelo de red, están organizadas dentro de dos grandes marcos de trabajo [4] – [5]:

- **SSF.OS**: encargado de modelar los dispositivos de red y sus sistemas operativos, especialmente los protocolos implementados en el escenario de red. Las principales clases de este paquete son:
 - ProtocolGraph
 - ProtocolSession
 - ProtocolMessage y PacketEvent.
- **SSF,Net**: encargado de modelar la conectividad de la red, crear y configurar nodos y enlaces; las principales clases de este paquete son:
 - Net.
 - Host y Router.
 - NIC.
 - Link.

Los anteriores marcos de trabajo, ocultan todos los detalles del simulador de eventos discretos SSF API, permitiendo implementar protocolos como si se ejecutaran en un sistema operativo real [4].

2.1. ARQUITECTURA Y CAPAS DEL SIMULADOR DE SSFNET.

La arquitectura de SSFNet se encuentra constituida por tres elementos esenciales para su funcionamiento, dichas elementos o capas son [5]:

- **DML** (*Domain Modeling Language*): Lenguaje de modelado de dominio, más conocido como el modelo de configuración de red.
- **SSFNet** (*SSF Network Models*): Modelo de Red SSF.
- **SSF** (*Scalable Simulation Framework*): Marco de Simulación Escalable.

Estas capas se pueden apreciar en la figura 1:

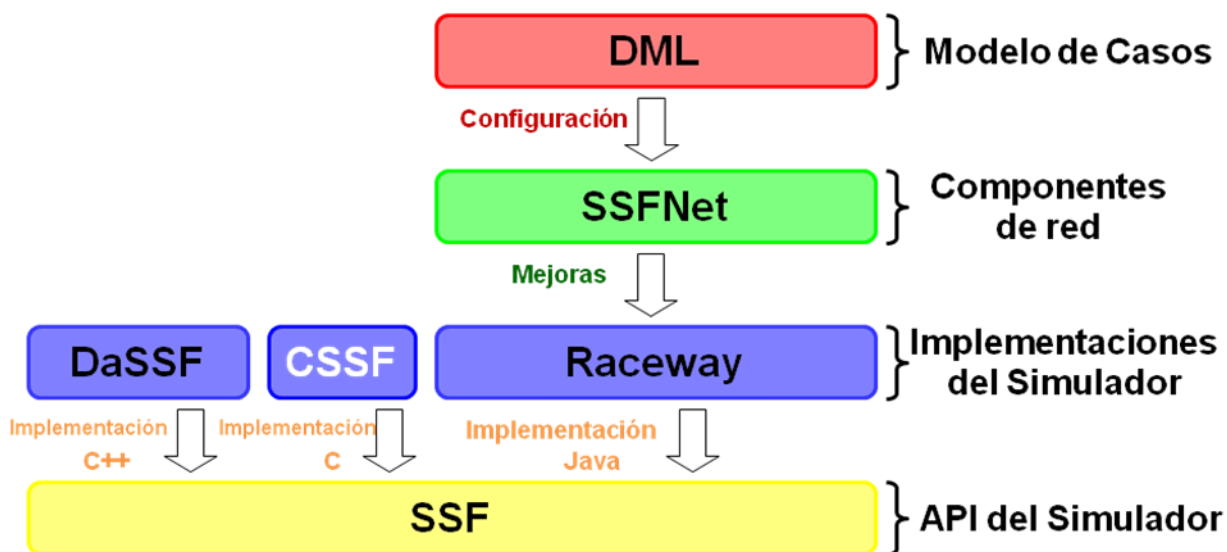


Figura 1 Capas de Simulación de SSFNet [5].

El orden de esta figura pretende presentar de una manera categórica y sencilla la elaboración del código y su posterior simulación; en primera instancia se crea el código DML que corresponde a los dominios, luego, el código relacionado con los elementos que operarán en la red, dispositivos, protocolos entre otras características importantes se definen en la capa SSFNet, y por último el API del simulador es el encargado de tomar todos estos códigos y realiza la respectiva simulación de eventos discretos.

2.2. DIRECCIONAMIENTO NHI EN SSFNET

La configuración del direccionamiento en el simulador SSFNet en lenguaje DML, es rigurosa y jerárquica, es decir, debe cumplir con ciertas pautas en su código, que permita ser entendida por cualquier proceso que se esté ejecutando; las siglas NHI corresponden a los elementos de red que componen un sistema autónomo de un escenario.

De tal forma que:

- **N** representa la identificación de la red o *Network Id*, es decir, para el AS1, N será 1, para el AS2, N será igual a 2, y así consecutivamente.
- **H** representa el identificador del equipo anfitrión o *Host Id*, de acuerdo al equipo que envía o recibe los datos.
- **I** representa el identificador de interface del Host, o *Interface Id*.

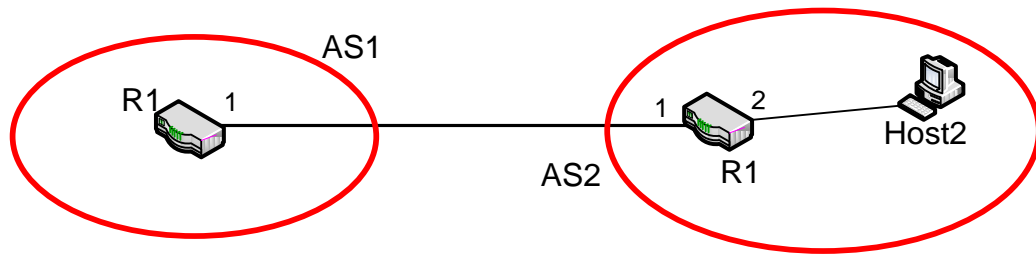


Figura 2 Enlace NHI entre dos enrutadores.

En el caso de la Figura 2, se necesita interconectar dos enrutadores de frontera de un AS1 y un AS2 como se muestra, la sentencia en lenguaje DML deberá ser:

Link [attach 1:1(1) attach 2:1(1)]

Asimismo deberá ser para la conexión entre el enrutador 1 y el host 2, ambos del AS2, de la siguiente forma:

Link [attach 2:1(2) attach 2:2(0)]

Donde la interfaz 0 es la interfaz predeterminada para la conexión de un terminal, el Host 2 en este caso.

3. CONFIGURACIÓN Y EJECUCIÓN DE UNA SIMULACIÓN EN SSFNET

Para configurar y ejecutar exitosamente un escenario de red en esta herramienta de simulación, se deben tener en cuenta una serie de pasos, que aun sin ser formales, pretenden establecer un orden entre la creación del modelo de red y la simulación final [4] - [6].

3.1. CREACIÓN DEL MODELO DE RED

Esta es la fase preliminar a la simulación, se deben tener diseñados los escenarios donde se harán las respectivas pruebas, el código puede ser elaborado en un bloc de notas pero debe ser guardado con extensión “.dml”; estos deberán especificar la topología de red, todos sus dominios junto a cada uno de los dispositivos que los constituirán, los enlaces entre ellos, protocolos que operarán y características propias del lenguaje de modelado de dominio, que son necesarias para el correcto funcionamiento de SSFNet,.

3.2. SSF RACEWAY

La capa Raceway que se observa en la figura 1, es el elemento adicional que se utilizó para la realización de las adaptaciones del protocolo BGP-4. Concretamente, la versión **ssfnet_raceway-2.0**, implementación que requiere el simulador para ejecutar sus comandos en este caso *Java*, debido a que emplea la programación multi-hilo que permite realizar simultáneamente un gran número de procesos y el uso de sockets que provee una comunicación entre los dispositivos del sistema.

Cabe anotar que la característica que hace mas fuerte a *ssfnet_raceway-2.0*, es la capacidad de realizar un seguimiento paso a paso de la simulación, aprovechándose de tal forma que se pueden monitorear las decisiones tomadas por el protocolo que ha sido adaptado, sin embargo, la interfaz de usuario no presenta la amabilidad al programador deseada, debido a que la configuración en su mayoría es escrita, a diferencia de otros simuladores que su accesibilidad radica en la opción de manejar el ratón únicamente.

3.3. EL COMPONENTE BGP

Dentro del código DML de cada uno de los escenarios, es preciso citar a la clase **SSF.OS.BGP4.BGPSession**, para utilizar el protocolo de enrutamiento BGP en la simulación.

El RFC 4271 establece el orden de ejecución de protocolos para el funcionamiento de BGP-4 en el proceso de comunicación, tal y como lo muestra la siguiente sección de código DML. [6].

```

router
[
  graph
  [
    ProtocolSession
    [
      name bgp
      use SSF.OS.BGP4.BGPSession
    ]
    ProtocolSession
    [
      name socket
      use SSF.OS.Socket.socketMaster
    ]
    ProtocolSession
    [
      name tcp
      use SSF.OS.TCP.tcpSessionMaster
    ]
    ProtocolSession
    [
      name ip
      use SSF.OS.IP
    ]
  ]
]

```

3.4. KIT DESARROLLADOR DE JAVA

Como se mencionaba en sección 2 de este mismo anexo, se debe instalar una herramienta llamada *Kit Desarrollador de Java* en su versión **1.2 (JDK: Java Development Kit)**, la cual consta de una serie de aplicaciones y componentes que permiten desarrollar nuevas aplicaciones, manipular y ejecutar códigos en lenguaje java.

Los componentes del JDK 1.2 se citan a continuación [7]:

- **Compilador:** se encarga de generar el *ByteCode*, que es un código intermedio entre el lenguaje del computador y *Java*, archivos con extensión “.class”, sin embargo este código no puede ejecutarse por sí mismo, en ninguna plataforma.
- **Interprete en tiempo de ejecución (JRE: Java Runtime Environment):** permite la ejecución de los archivos con extensión “.class”, contiene un conjunto de bibliotecas de *Java*, una Máquina Virtual de Java (JVM: Java Virtual Machine) además de otros componentes necesarios para que pueda intermediar entre el sistema operativo y el lenguaje, y finalmente es el que convierte el código Java en un código máquina o archivo fuente que es capaz de ejecutarse en cualquier plataforma, en lo que radica la fuerza de las aplicaciones en Java.
- **Visualizador de applets¹:** es la herramienta cuya función es servir como campo de pruebas de los applets, permitiendo visualizar los archivos fuente en un navegador, sin necesidad de esperar a tener una aplicación completa para evaluar su comportamiento.
- **Depurador:** permite encontrar errores y eliminarlos, pero sin características visuales.

Todos los componentes del JDK, son necesarios para que sean implementados por el SSF Raceway, desde un lenguaje Java que pueda interactuar con los escenarios especificados en lenguaje DML.

3.5. CONFIGURACIÓN Y EJECUCIÓN DEL SIMULADOR.

Siguiendo en orden los anteriores pasos, con su respectiva instalación, creación de escenarios y demás, resta determinar el tiempo de simulación de cada escenario, el valor es arbitrario, depende directamente de qué tan grande es el escenario, de cuánta información necesitemos intercambiar, entre otros factores que permitan obtener unos resultados coherentes y esperados.

En líneas de código, la configuración del simulador junto con el tiempo de simulación deseado es la siguiente [4]:

- `set CLASSPATH=C:\ssfnet\lib\raceway.jar;C:\ssfnet\lib\ssfnet.jar`
- `set CLASSPATH=%CLASSPATH%;C:\ssfnet\lib\cernlite.jar`
- `set CLASSPATH=%CLASSPATH%;C:\ssfnet\lib\raceway.jar`
- `set CLASSPATH=%CLASSPATH%;C:\ssfnet\lib\regexp.jar`
- `set CLASSPATH=%CLASSPATH%;C:\ssfnet\src\`
- `java SSF.Net.Net 1234 myBigModel.dml C:\ssfnet\examples\net.dml`

Se debe tener en cuenta que la carpeta del simulador SSFNet, debe estar en el disco C:, en este caso el mismo en donde se encuentra el sistema operativo y la instalación del JDK 1.2.

¹ Componente de una aplicación, en este caso de Java, que se ejecuta independientemente de las demás partes de la aplicación, visualizada en un navegador para efectos de pruebas, cumpliendo una función muy específica; sin embargo, un applet por sí solo no podría servir como una aplicación como tal [8].

El valor “1234”, es el tiempo en segundos definido para la simulación, para el presente proyecto, el tiempo en segundos es de 20.000, siendo un tiempo significativo para observar con detalle el comportamiento de la información por los escenarios además que, la ruta *C:\ssfnetwork\examples\net.dml*, es el esquema oficial SSFNet del archivo DML que se utiliza para la validación automática del archivo de entrada DML, *bgp.dml*.

Al comienzo de la simulación, el objeto *SSF.Net.Net* utiliza los servicios de la biblioteca DML para cargar el contenido del archivo *bgp.dml* y *net.dml* en una base de datos de configuración en tiempo de ejecución.

Posteriormente, *SSF.Net.Net* configura de manera sistemática todos los objetos de simulación (terminales, enrutadores, protocolos, redes y enlaces). Una vez que todos los objetos de simulación se han instanciado, la fase de inicialización comienza llamando al método *init ()*; para finalmente invocar al método *startAll ()*, con lo cual la simulación comienza 0 segundos.

3.6. RESULTADOS DEL SIMULADOR.

Al finalizar los pasos propuestos anteriormente, luego de seguir las recomendaciones y determinar el tiempo de simulación deseado, se debe ejecutar un comando en la consola de Windows que emula el entorno DOS, que primero se ubica en la ruta donde se encuentra localizado la sección de pruebas del simulador SSFNet, dentro de la zona del protocolo BGP4, así:

```
C:\> cd C:\ssfnetwork\src\SSF\OS\BGP4\test\BGPC2
```

Luego, desde esta ubicación, se debe ejecutar el siguiente comando:

```
java SSF.Net.Net 20000 nombre_archivo.dml ..\dictionary.dml > nom_resultado.out 2>&1
```

Donde, mencionado antes:

- *20000* es el tiempo de simulación en segundos.
- *nombre_archivo.dml*, es el archivo DML de un escenario, el cual servirá para simular el protocolo BGPC2.
- *nom_resultado.out*, es el nombre que el usuario asigna al resultado que entrega el simulador.

El simulador entrega un archivo *.out*, allí están consignadas las tablas de enrutamiento de los dispositivos BGP, a nivel inter-dominio, los paquetes y el total de información en bytes que se entrega de origen a destino entre mucha otra información importante para diferentes estudios.

El archivo *.out*, ofrece información importante para esta investigación, en las siguientes imágenes, se pueden observar los resultados arrojados por el simulador, en un archivo llamado *E3_modif_C0.out*.

La primera de ellas (ver *Figura 3*) muestra el número de paquetes recibidos por el AS destino, junto con su tamaño en bytes, el tiempo en el que se creó el paquete, el tiempo que tardo en llegar al destino y el tiempo total entre el origen y el destino, además de información relevante para otros aspectos.

```

264.932486861 [ sid 0 start 208.902776294 ] cInt 1:3 srv 2:3(0) #pages: 2 #objects: 2 total: 10931B seconds: 56.029710567 SUCCESS
367.161446375 [ sid 1 start 296.242329394 ] cInt 1:3 srv 2:3(0) #pages: 3 #objects: 5 total: 55443B seconds: 70.919116981 SUCCESS
1189.823907206 [ sid 2 start 404.403446998 ] cInt 1:3 srv 2:3(0) #pages: 18 #objects: 48 total: 206334B seconds: 785.420460208 SUCCESS
1216.927855778 [ sid 3 start 1215.409762291 ] cInt 1:3 srv 2:3(0) #pages: 1 #objects: 6 total: 60264B seconds: 1.518093487 SUCCESS
2007.165556667 [ sid 4 start 123.712269762 ] cInt 1:3 srv 2:3(0) #pages: 17 #objects: 32 total: 159035B seconds: 783.453286905 SUCCESS
2550.733948907 [ sid 5 start 2075.793307567 ] cInt 1:3 srv 2:3(0) #pages: 9 #objects: 15 total: 77848B seconds: 474.94064134 SUCCESS
2742.787857407 [ sid 6 start 2650.374150656 ] cInt 1:3 srv 2:3(0) #pages: 4 #objects: 7 total: 22907B seconds: 92.413706751 SUCCESS
2918.85834684 [ sid 7 start 2918.850069979 ] cInt 1:3 srv 2:3(0) #pages: 1 #objects: 1 total: 2307B seconds: 0.008276861 SUCCESS
2986.252495711 [ sid 8 start 2986.244063973 ] cInt 1:3 srv 2:3(0) #pages: 1 #objects: 1 total: 2510B seconds: 0.008431738 SUCCESS
3573.094898104 [ sid 9 start 3475.845953883 ] cInt 1:3 srv 2:3(0) #pages: 4 #objects: 4 total: 23095B seconds: 99.248947223 SUCCESS
3823.647603749 [ sid 10 start 3584.996547791 ] cInt 1:3 srv 2:3(0) #pages: 6 #objects: 7 total: 32423B seconds: 238.651055958 SUCCESS
4037.534299263 [ sid 11 start 3863.874154015 ] cInt 1:3 srv 2:3(0) #pages: 6 #objects: 11 total: 70697B seconds: 173.680145248 SUCCESS
4427.963214553 [ sid 12 start 4189.932587986 ] cInt 1:3 srv 2:3(0) #pages: 7 #objects: 8 total: 38679B seconds: 238.030265677 SUCCESS
4802.794511456 [ sid 13 start 4499.548572672 ] cInt 1:3 srv 2:3(0) #pages: 7 #objects: 7 total: 142388B seconds: 303.245938784 SUCCESS
5034.450736315 [ sid 14 start 5034.441380657 ] cInt 1:3 srv 2:3(0) #pages: 1 #objects: 1 total: 3721B seconds: 0.009355658 SUCCESS
5047.866648681 [ sid 15 start 5047.855237655 ] cInt 1:3 srv 2:3(0) #pages: 1 #objects: 1 total: 5775B seconds: 0.011991026 SUCCESS
5346.754458093 [ sid 16 start 5346.744052631 ] cInt 1:3 srv 2:3(0) #pages: 1 #objects: 1 total: 5057B seconds: 0.010405462 SUCCESS
5522.510593706 [ sid 17 start 5489.311742614 ] cInt 1:3 srv 2:3(0) #pages: 2 #objects: 9 total: 38978B seconds: 33.198851092 SUCCESS
5718.437930294 [ sid 18 start 5537.295510578 ] cInt 1:3 srv 2:3(0) #pages: 5 #objects: 7 total: 41612B seconds: 181.142419716 SUCCESS
6326.648978839 [ sid 19 start 5764.998019045 ] cInt 1:3 srv 2:3(0) #pages: 13 #objects: 77 total: 489383B seconds: 561.650959794 SUCCESS
6678.089264947 [ sid 20 start 6423.551582154 ] cInt 1:3 srv 2:3(0) #pages: 8 #objects: 34 total: 1394852B seconds: 254.537682793 SUCCESS
7445.398552657 [ sid 21 start 7043.758456045 ] cInt 1:3 srv 2:3(0) #pages: 7 #objects: 29 total: 169536B seconds: 401.640099612 SUCCESS
7853.775436234 [ sid 22 start 7791.212298679 ] cInt 1:3 srv 2:3(0) #pages: 3 #objects: 4 total: 202043B seconds: 62.563137444 SUCCESS
7950.069590188 [ sid 23 start 7950.050475477 ] cInt 1:3 srv 2:3(0) #pages: 1 #objects: 1 total: 15450B seconds: 0.019114711 SUCCESS
8283.59253476 [ sid 24 start 8065.153132757 ] cInt 1:3 srv 2:3(0) #pages: 7 #objects: 26 total: 520140B seconds: 218.439402003 SUCCESS
8332.394308953 [ sid 25 start 8332.190976103 ] cInt 1:3 srv 2:3(0) #pages: 1 #objects: 2 total: 5544B seconds: 0.203328255 SUCCESS
8475.168418918 [ sid 26 start 8403.200233057 ] cInt 1:3 srv 2:3(0) #pages: 3 #objects: 4 total: 14470B seconds: 71.968185861 SUCCESS
9171.647939758 [ sid 27 start 8892.337335082 ] cInt 1:3 srv 2:3(0) #pages: 7 #objects: 10 total: 49809B seconds: 279.310604676 SUCCESS
9308.866897844 [ sid 28 start 9172.806490415 ] cInt 1:3 srv 2:3(0) #pages: 3 #objects: 4 total: 51138B seconds: 136.060407429 SUCCESS
9416.147036889 [ sid 29 start 9371.32461078 ] cInt 1:3 srv 2:3(0) #pages: 2 #objects: 6 total: 32369B seconds: 44.822422909 SUCCESS
10145.697978895 [ sid 30 start 9717.084426504 ] cInt 1:3 srv 2:3(0) #pages: 9 #objects: 21 total: 212702B seconds: 428.613352391 SUCCESS
10426.84235864 [ sid 31 start 10355.306794689 ] cInt 1:3 srv 2:3(0) #pages: 3 #objects: 5 total: 31825B seconds: 71.535563951 SUCCESS
10733.882344339 [ sid 32 start 10494.721081869 ] cInt 1:3 srv 2:3(0) #pages: 6 #objects: 6 total: 61588B seconds: 239.16126247 SUCCESS
11213.945665834 [ sid 33 start 10772.036499031 ] cInt 1:3 srv 2:3(0) #pages: 11 #objects: 28 total: 135744B seconds: 441.909166803 SUCCESS
11295.230808436 [ sid 34 start 11261.744346747 ] cInt 1:3 srv 2:3(0) #pages: 2 #objects: 2 total: 13734B seconds: 33.486461689 SUCCESS
11456.40890101 [ sid 35 start 11352.122762565 ] cInt 1:3 srv 2:3(0) #pages: 4 #objects: 15 total: 279592B seconds: 104.286138445 SUCCESS
11598.713039462 [ sid 36 start 11530.058970058 ] cInt 1:3 srv 2:3(0) #pages: 3 #objects: 5 total: 129799B seconds: 68.654069404 SUCCESS
11725.158261386 [ sid 37 start 11698.513477767 ] cInt 1:3 srv 2:3(0) #pages: 2 #objects: 2 total: 23575B seconds: 26.644783619 SUCCESS
12945.520448907 [ sid 38 start 11819.693852463 ] cInt 1:3 srv 2:3(0) #pages: 22 #objects: 83 total: 459043B seconds: 1125.826596444 SUCCESS
13060.449100239 [ sid 39 start 12991.769465145 ] cInt 1:3 srv 2:3(0) #pages: 3 #objects: 5 total: 23047B seconds: 68.679635074 SUCCESS
13064.041042319 [ sid 40 start 13062.382764391 ] cInt 1:3 srv 2:3(0) #pages: 1 #objects: 8 total: 56494B seconds: 2.279798254 SUCCESS
13304.290195193 [ sid 41 start 13237.689948289 ] cInt 1:3 srv 2:3(0) #pages: 2 #objects: 2 total: 5724B seconds: 66.600246904 SUCCESS
13376.146677677 [ sid 42 start 13339.264745868 ] cInt 1:3 srv 2:3(0) #pages: 2 #objects: 2 total: 6086B seconds: 36.88392019 SUCCESS
13980.332370482 [ sid 43 start 13474.507989718 ] cInt 1:3 srv 2:3(0) #pages: 8 #objects: 49 total: 306349B seconds: 505.824380764 SUCCESS
14002.680585995 [ sid 44 start 14002.397019112 ] cInt 1:3 srv 2:3(0) #pages: 1 #objects: 2 total: 35439B seconds: 0.283566883 SUCCESS
14120.266328822 [ sid 45 start 14120.253308754 ] cInt 1:3 srv 2:3(0) #pages: 1 #objects: 1 total: 5544B seconds: 0.011020068 SUCCESS
14169.236677515 [ sid 46 start 14166.956878689 ] cInt 1:3 srv 2:3(0) #pages: 1 #objects: 8 total: 56494B seconds: 2.279798254 SUCCESS
14377.243232606 [ sid 47 start 14172.354377745 ] cInt 1:3 srv 2:3(0) #pages: 4 #objects: 11 total: 57224B seconds: 204.888854861 SUCCESS
14668.165914769 [ sid 48 start 14419.952632296 ] cInt 1:3 srv 2:3(0) #pages: 4 #objects: 7 total: 24092B seconds: 248.213281809 SUCCESS
15299.778369142 [ sid 49 start 14670.10362976 ] cInt 1:3 srv 2:3(0) #pages: 12 #objects: 17 total: 118474B seconds: 629.674739382 SUCCESS
16382.287885841 [ sid 50 start 15363.695191124 ] cInt 1:3 srv 2:3(0) #pages: 23 #objects: 42 total: 396711B seconds: 1018.592694717 SUCCESS
16437.667393158 [ sid 51 start 16393.020830461 ] cInt 1:3 srv 2:3(0) #pages: 2 #objects: 2 total: 38347B seconds: 44.646562697 SUCCESS
16471.370892157 [ sid 52 start 16470.766081448 ] cInt 1:3 srv 2:3(0) #pages: 3 #objects: 3 total: 58702B seconds: 0.604810709 SUCCESS
16615.442013106 [ sid 53 start 16556.035816467 ] cInt 1:3 srv 2:3(0) #pages: 2 #objects: 2 total: 9069B seconds: 59.406196639 SUCCESS
16739.865836344 [ sid 54 start 16739.84589537 ] cInt 1:3 srv 2:3(0) #pages: 1 #objects: 1 total: 16493B seconds: 0.019940974 SUCCESS
16791.895207988 [ sid 55 start 16759.495303742 ] cInt 1:3 srv 2:3(0) #pages: 2 #objects: 5 total: 26717B seconds: 32.399904246 SUCCESS
16971.129523695 [ sid 56 start 16971.121255989 ] cInt 1:3 srv 2:3(0) #pages: 1 #objects: 1 total: 22958B seconds: 0.008267706 SUCCESS
17311.317126706 [ sid 57 start 17028.103776163 ] cInt 1:3 srv 2:3(0) #pages: 8 #objects: 35 total: 293058B seconds: 283.213350543 SUCCESS
17501.718736515 [ sid 58 start 17459.282568773 ] cInt 1:3 srv 2:3(0) #pages: 2 #objects: 4 total: 135769B seconds: 42.436167742 SUCCESS
17607.674147825 [ sid 59 start 17607.069209827 ] cInt 1:3 srv 2:3(0) #pages: 1 #objects: 2 total: 11415B seconds: 0.604937988 SUCCESS
17755.300848812 [ sid 60 start 17754.901456958 ] cInt 1:3 srv 2:3(0) #pages: 1 #objects: 2 total: 4746B seconds: 0.399391854 SUCCESS
17998.570382659 [ sid 61 start 17972.019408956 ] cInt 1:3 srv 2:3(0) #pages: 2 #objects: 2 total: 10341B seconds: 26.550973703 SUCCESS
18186.361945847 [ sid 62 start 18157.663677365 ] cInt 1:3 srv 2:3(0) #pages: 2 #objects: 2 total: 24971B seconds: 28.698268482 SUCCESS
18449.675239206 [ sid 63 start 18365.998692201 ] cInt 1:3 srv 2:3(0) #pages: 2 #objects: 2 total: 18458B seconds: 83.676547005 SUCCESS
19281.526510232 [ sid 64 start 18572.354343446 ] cInt 1:3 srv 2:3(0) #pages: 13 #objects: 29 total: 132978B seconds: 709.172175886 SUCCESS

```

Figura 3 Número de paquetes entregados.

Posteriormente, el archivo muestra (ver *Figura 4*) las tablas de enrutamiento de cada uno de los equipos BGP dentro de un AS, detallando número de saltos, cuales dominios deben cruzar para llegar al destino, entre otros datos importantes, así:

```

.....
.....|..... bgp@1:1 wrap-up .....
.....
~# 1:1 --- Loc-RIB at bgp@1:1:
~# 1:1 | NetworkNHI NextHopNHI Metric LocPrf Weight ASPathNHI
~# 1:1 | *> 2 1:2 {1} - 100 - 2 i
~# 1:1 | *> 1 self - - - - i
.....
..... bgp@1:2 wrap-up .....
.....
~# 1:2 --- Loc-RIB at bgp@1:2:
~# 1:2 | NetworkNHI NextHopNHI Metric LocPrf Weight ASPathNHI
~# 1:2 | *> 2 2:2 {2} - 100 - 2
~# 1:2 | *> 1 self - - - - i
.....

```

Figura 4 Tablas de enrutamiento de equipos BGP.

Sin embargo, SSFNet presenta varias limitaciones, debido a que este simulador solo entrega información inter-dominio, no presenta información relevante que se presenta a nivel intra-dominio, es de una herramienta de eventos discretos, es decir, una vez iniciada la simulación, los características de la red no cambian, si se necesita algún cambio, se debe esperar a que termine la simulación, realizar los cambios pertinentes e iniciar de nuevo la simulación, sin embargo, los resultados que entrega en su archivo *.out*, son bastantes completos.

4. ESCENARIOS IMPLEMENTADOS, CÓDIGOS DML.

Es importante recordar que los escenarios utilizados para esta investigación, fueron tomados del trabajo de grado, Estudio de Viabilidad para la Optimización de Enrutamiento IP con el Protocolo BGP, en donde se evaluó el comportamiento de J2-BGP², debido a que son idóneos en su representación de las diferentes redes que podrían presentarse y servirán de gran ayuda en el análisis de la adaptación BGPC2 aquí propuesta.

4.1. ESCENARIO 1

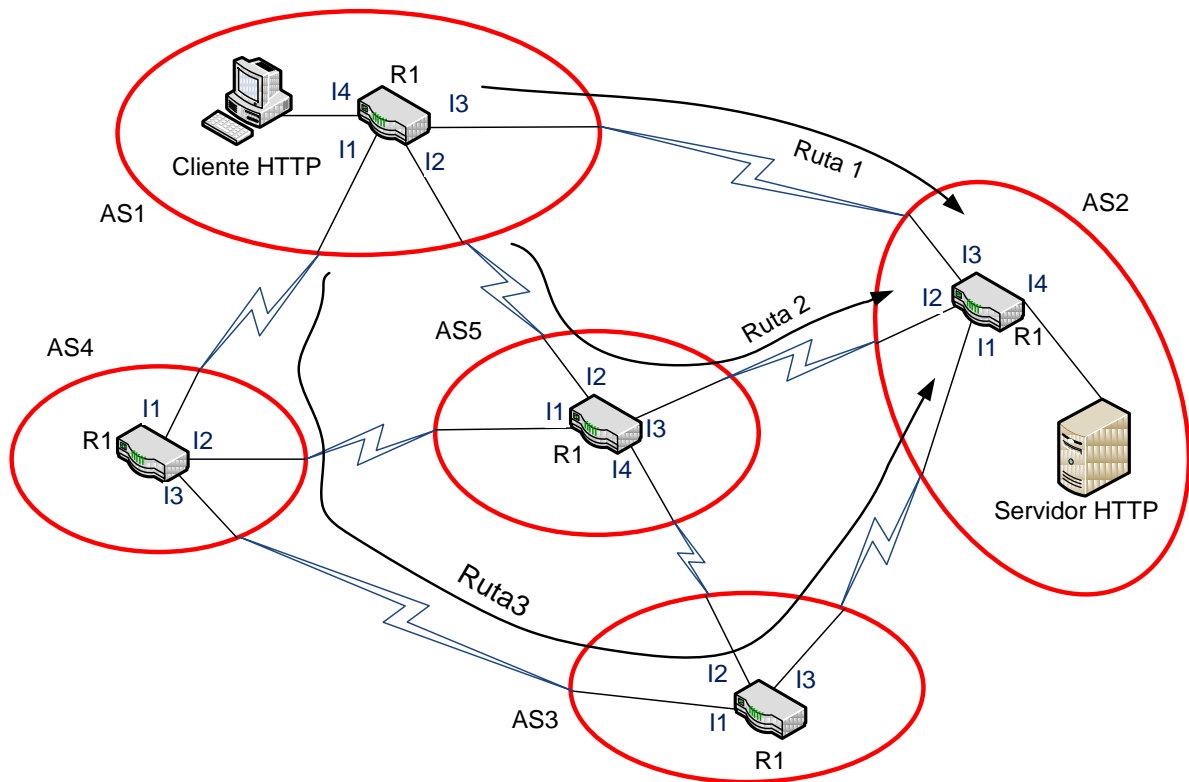


Figura 5 Escenario de simulación 1.

² J2-BGP, modificación al protocolo BGP-4, propuesta en el trabajo de grado “Estudio de Viabilidad para la Optimización de Enrutamiento IP con el Protocolo BGP”, cuyo aporte fue evaluar una ruta dependiendo al retardo que esta presentaba, y así seleccionar la que presentaba un menor retardo en sus enlaces [9].

```

_schema
[
  _find .schemas.Net          # Busca los diseños generales para enrutadores y host del archivo
                              # net.dml.
]

Net                            # Net es el atributo de mayor nivel que encierra toda la
                              # Configuración de la red.

[
  frequency 1000000000        # Resolución de la simulación en términos de nanosegundos 10^-9 seg.
                              # Es decir que 1 tick=1 nanosegundo, lo que significa que el tiempo de
                              # simulación avanzará en intervalos múltiplos de nanosegundos.

randomstream
[
  generator "MersenneTwister"
  stream "seedstarter2"
  reproducibility_level "host"
]

traffic
[
  pattern
  [
    client 1:2
    servers
    [
      list "foobar" nhi 2:2(0) port 80
    ]
  ]
]

bgpoptions                      # El atributo bgpoptions se utiliza con propósitos de diagnóstico e
                              # Información de la simulación; consiste # una lista de atributos DML
                              # que se usan para depurar y validar los resultados de simulación.

[

validation_test 12             # Imprime mensajes asociados con una prueba de validación en
                              # particular, en este caso la número 12.

startup_jitter_bound 0.1
  # show_id_data true
  # show_nb_info true
  # show_snd_open true
  # show_snd_up true
  # show_in_policy true
  # show_dop_calc true
  # show_dec_proc true
  # show_fwd_tables true
  # show_out_policy true
  # dump_fwd_tables true
  # dump_ribs_in true

```

```

# dump_stability true
# dump_ribs_out true
# show_loc_rib true
# dump_loc_rib true
# show_ribs_out true
# show_loc_rib true
]

Net # Se identifica a una red dentro de la topología.
[
id 5 # El identificador de la red es 5.
AS_status boundary # Significa que esta red encierra un Sistema Autónomo
router # Especifica que en esta red existe un enrutador.
[
id 1 # El identificador del enrutador es 1.
graph # graph es un atributo que especifica una lista de protocolos que
# pueden ser configurados dentro del enrutador y los nombres de las
# clases del paquete SSF.OS que se van a usar.

[
ProtocolSession # Etiqueta que especifica el protocolo que va a utilizar el enrutador.
[
name bgp # En este caso para la capa de aplicación se utiliza BGP.
use SSF.OS.BGP4.BGPSession # Significa que va a utilizar la clase BGPSession del paquete
# SSF.OS.BGP4.

autoconfig false # Esto significa que la configuración automática esta desactivada,
# por lo cual deben especificarse todos los parámetros del enrutador.

connretry_time 120 # Significa que deben transcurrir 120 seg. antes de intentar reiniciar
# un intento fallido de conexión con un par BGP.

min_as_orig_time 15 # Significa que el tiempo mínimo que debe pasar entre anuncios
# sucesivos y mensajes Update enviados debe ser de 15 segs.
# Desde este enrutador y que reportan cambios dentro del AS local.

reflector false # La característica de route reflector esta desactivada.
neighbor # Encapsula los atributos de un vecino BGP.
[
as 1 # El identificador del AS vecino es 1.
address 1(2) # Dirección de la interfaz NHI relativa al AS vecino. En este caso
# 1(2) significa Enrutador 1(Interfaz 2).

use_return_address 1(2) # Dirección de la interfaz NHI relativa al AS local que el vecino
# debe usar cuando envíe mensajes al enrutador BGP local.
# En este caso debe usar la interfaz 2 del enrutador 1

_extends .basic_ebgp_neighbor # Significa que hereda los otros parámetros para el enrutador del
# método basic_ebgp_neighbor presente en el archivo dictionary.dml
]
neighbor
[
as 2 # El identificador del AS vecino es 2.
address 1(2) # Dirección de la interfaz NHI relativa al AS vecino. En este caso
# 1(2) significa Enrutador 1(Interfaz 2).

use_return_address 1(3) # Dirección de la interfaz NHI relativa al AS local que el vecino
# debe usar cuando envíe mensajes al enrutador BGP local.
# En este caso debe usar la interfaz 3 del enrutador 1. Es decir
# Enrutador 1(Interfaz 3).

_extends .basic_ebgp_neighbor # Significa que hereda los otros parámetros para el enrutador del
# método basic_ebgp_neighbor presente en el archivo dictionary.dml
]
]
]

```

```

    ]
neighbor
  [
    as 3 # El identificador del AS vecino es 3.
    address 1(2) # Dirección de la interfaz NHI relativa al AS vecino. En este caso
                # 1(2) significa Enrutador 1(Interfaz 2).
    use_return_address 1(4) # Dirección de la interfaz NHI relativa al AS local que el vecino
                            # debe usar cuando envíe mensajes al enrutador BGP local.
                            # En este caso debe usar la interfaz 4 del enrutador 1. Es decir
                            # Enrutador 1(Interfaz 4).
    _extends .basic_ebgp_neighbor # Significa que hereda los otros parámetros para el enrutador
                                  # del método basic_ebgp_neighbor presente en el archivo dictionary.dml
  ]
neighbor
  [
    as 4 # El identificador del AS vecino es 4.
    address 1(2) # Dirección de la interfaz NHI relativa al AS vecino. En este caso
                # 1(2) significa Enrutador 1(Interfaz 2).
    use_return_address 1(1) # Dirección de la interfaz NHI relativa al AS local que el vecino
                            # debe usar cuando envíe mensajes al enrutador BGP local.
                            # En este caso debe usar la interfaz 1 del enrutador 1.
    _extends .basic_ebgp_neighbor # Significa que hereda los otros parámetros para el enrutador del
                                  # método basic_ebgp_neighbor presente en el archivo dictionary.dml
  ]
monitor # Encapsula atributos para configurar el muestreo de datos respectivos
        #a este enrutador.
  [
    # show_in_policy true
    dump_zmrt_updates enrutador1 # Arroja información del enrutador cada vez que se
                                  # producen actualizaciones en un archivo tipo MRT llamado enrutador1.
  ]
]
ProtocolSession # Etiqueta que especifica el protocolo que va a utilizar el enrutador.
  [
    name socket # En este caso socket.
    use SSF.OS.Socket.socketMaster # Significa que va a utilizar la clase socketMaster del paquete
                                    # SSF.OS.Socket.
  ]
ProtocolSession # Etiqueta que especifica el protocolo que va a utilizar el
enrutador.
  [
    name tcp # En este caso para la capa de transporte se utiliza TCP.
    use SSF.OS.TCP.tcpSessionMaster # Significa que va a utilizar la clase tcpSessionMaster
                                     # del paquete SSF.OS.TCP.
    warn false # Desactiva el muestreo de las advertencias o Warnings.
  ]
ProtocolSession # Etiqueta que especifica el protocolo que va a utilizar el
# enrutador.
  [
    name ip # En este caso para la capa de red se utiliza IP.
    use SSF.OS.IP # Significa que va a utilizar el paquete SSF.OS.IP.
  ]
]
interface # Declaración de las interfaces.

```



```

    [
      id 0                                # Identificador de las interfaces, en este caso la Interfaz 0
                                          # correspondiente a la interfaz de loopback.
      virtual true                        # La interfaz de loopback se especifica como una interfaz virtual.
    ]
  interface                              # Declaración de las interfaces.
  [
    idrange                               # Especifica el rango de interfaces que posee el enrutador.
    [
      from 1 to 4                         # En este caso las interfaces 1, 2, 3 y 4.
    ]
  ]
]
]

Net                                       # Se declara una red dentro de la topología.
[
  id 1                                    # El identificador de la red es 1.
  AS_status boundary                     # Significa que esta red encierra un Sistema Autónomo, o lo que es lo
                                          # mismo que esta red es el AS 1.
  router                                 # Especifica que en esta red existe un enrutador.
  [
    id 1                                  # El identificador del enrutador es 1.
    graph                                 # graph es un atributo que especifica una lista de protocolos que
                                          # pueden ser configurados dentro del enrutador y los nombres de las
                                          # clases del paquete SSF.OS que se van a usar.

    [
      ProtocolSession                    # Etiqueta que especifica el protocolo que va a utilizar el enrutador.
      [
        name bgp
        use SSF.OS.BGP4.BGPSession      # Significa que va a utilizar la clase BGPSession del paquete
                                          # SSF.OS.BGP4.
        autoconfig false                 # Esto significa que la configuración automática esta desactivada,
                                          # por lo cual deben especificarse todos los parámetros del enrutador.
        connretry_time 120              # Significa que deben transcurrir 120 seg. antes de intentar reiniciar
                                          # un intento fallido de conexión con un par BGP.
        min_as_orig_time 15             # Significa que la cantidad mínima de tiempo que debe pasar entre
                                          # anuncios sucesivos e mensajes Update enviados.
        reflector false                  # La característica de route reflector esta desactivada.
        neighbor                          # Encapsula los atributos de un vecino BGP.
        [
          as 5                            # El identificador del AS vecino es 5.
          address 1(2)                    # Dirección de la interfaz NHI relativa al AS vecino. En este caso
                                          # 1(2) significa Enrutador 1(Interfaz 2).
          use_return_address 1(2)        # Dirección de la interfaz NHI relativa al AS local que el vecino
                                          # debe usar cuando envíe mensajes al enrutador BGP local.
                                          # En este caso debe usar la interfaz 2 del enrutador 1. Es decir
                                          # Enrutador 1(Interfaz 2).
          hold_time 90                    # Tiempo máximo de espera en seg para que se pierda conectividad
                                          # entre los pares BGP si no se recibe un mensaje Update o Keep Alive.
          keep_alive_time 30              # Intervalo de tiempo en seg para el envío de mensajes KeepAlive.
          mrai 30                          # Mínima cantidad de tiempo en segundos que debe transcurrir entre
                                          # anuncios sucesivos de rutas.
        ]
      ]
    ]
  ]
]

```

```

infilter                                     # Encapsula las políticas de filtrado para los anuncios de rutas
                                             # provenientes desde este vecino.
  [
    clause                                   # Definición de una de las cláusulas que determinará las políticas de
                                             # filtrado.
      [
        precedence 1                       # Especifica el orden en que serán evaluadas las cláusulas cuando
                                             # inicie la simulación, en este caso la número 1.
        predicate []                       # Especifica un predicado asociado a la cláusula correspondiente, el
                                             # hecho de que este vacío significa que todas las rutas cumplen con
                                             # Este predicado.
        action                             # Acción que va a aplicarse a la ruta a través de esta cláusula.
          [
            primary permit                 # Acción primaria a ser aplicada en la ruta, esta puede ser permitir
                                             # o denegar la acción.
          ]
      ]
    ]
  ]
outfilter                                   # Encapsula las políticas de filtrado para los anuncios de rutas con
                                             # rumbo a este vecino.
  [
    _extends .filters.permit_all          # Hereda del método filters ubicado en el archivo dictionary.dml y el
                                             # cual permite todas las acciones sobre la ruta; es decir que no
                                             # posee políticas de filtrado.
  ]
neighbor                                    # Encapsula los atributos de un vecino BGP.
  [
    as 2                                    # El identificador del AS vecino es 2.
    address 1(3)                          # Dirección de la interfaz NHI relativa al AS vecino. En este caso
                                             # 1(3) significa Enrutador 1(Interfaz 3).
    use_return_address 1(3)              # Dirección de la interfaz NHI relativa al AS local que el vecino
                                             # debe usar cuando envíe mensajes al enrutador BGP local.
                                             # En este caso debe usar la interfaz 3 del enrutador 1. Es decir
                                             # Enrutador 1(Interfaz 3).
    hold_time 90                          # Tiempo máximo de espera en seg para que se pierda conectividad
                                             # entre los pares BGP si no se recibe un mensaje Update o Keep Alive.
    keep_alive_time 30                   # Intervalo de tiempo en seg para el envío de mensajes KeepAlive.
    mrai 30                               # Mínima cantidad de tiempo en segundos que debe transcurrir entre
                                             # anuncios sucesivos de rutas.
    infilter                              # Encapsula las políticas de filtrado para los anuncios de rutas
                                             # provenientes desde este vecino.
      [
        clause                             # Definición de una de las cláusulas que determinará las políticas de
                                             # filtrado.
          [
            precedence 1                   # Especifica el orden en que serán evaluadas las cláusulas cuando
                                             # inicie la simulación, en este caso la número 1.
            predicate []                   # Especifica un predicado asociado a la cláusula correspondiente, el
                                             # hecho de que este vacío significa que todas las rutas cumplen con
                                             # Este predicado.
            action                         # Acción que va a aplicarse a la ruta a través de esta cláusula.
              [
                primary permit             # Acción primaria a ser aplicada en la ruta, esta puede ser permitir
                                             # o denegar la acción.
              ]
          ]
      ]
  ]

```

```

    ]
  ]
]
outfilter
    # Encapsula las políticas de filtrado para los anuncios de rutas con
    # rumbo a este vecino.
    [
      _extends .filters.permit_all # Hereda del método filters ubicado en el archivo dictionary.dml y el
      # cual permite todas las acciones sobre la ruta; es decir que no
      # posee políticas de filtrado.
    ]
neighbor
    # Encapsula los atributos de un vecino BGP.
    [
      as 4 # El identificador del AS vecino es 4.
      address 1(1) # Dirección de la interfaz NHI relativa al AS vecino. En este caso
      # 1(1) significa Enrutador 1(Interfaz 1).
      use_return_address 1(1) # Dirección de la interfaz NHI relativa al AS local que el vecino
      # debe usar cuando envíe mensajes al enrutador BGP local.
      # En este caso debe usar la interfaz 1 del enrutador 1. Es decir
      # Enrutador 1(Interfaz 1).
      hold_time 90 # Tiempo máximo de espera en seg para que se pierda conectividad
      # entre los pares BGP si no se recibe un mensaje Update o Keep Alive.
      keep_alive_time 30 # Intervalo de tiempo en seg para el envío de mensajes KeepAlive.
      mrai 30 # Mínima cantidad de tiempo en segundos que debe transcurrir entre
      # anuncios sucesivos de rutas.
      infilter
        # Encapsula las políticas de filtrado para los anuncios de rutas
        # provenientes desde este vecino.
        [
          clause
            # Permitir todas las rutas aprendidas desde el AS 4.
            # Definición de una de las cláusulas que determinará las políticas de
            # filtrado.
            [
              precedence 1 # Especifica el orden en que serán evaluadas las cláusulas cuando
              # inicie la simulación, en este caso la número 1.
              predicate [] # Especifica un predicado asociado a la cláusula correspondiente, el
              # hecho de que este vacío significa que todas las rutas cumplen con
              # Este predicado.
              action
                # Acción que va a aplicarse a la ruta a través de esta cláusula.
                [
                  primary permit # Permitir todas las rutas aprendidas desde el AS 4.
                ]
            ]
          ]
        ]
      outfilter
        # Encapsula las políticas de filtrado para los anuncios de rutas con
        # rumbo a este vecino.
        [
          _extends .filters.permit_all # Hereda del método filters ubicado en el archivo dictionary.dml y el
          # cual permite todas las acciones sobre la ruta; es decir que no
          # posee políticas de filtrado.
        ]
      ]
    ]
ProtocolSession
    # Etiqueta que especifica el protocolo que va a utilizar el enrutador.
    [
      name socket # En este caso socket.
    ]

```

```

    use SSF.OS.Socket.socketMaster # Significa que va a utilizar la clase socketMaster del paquete
                                   # SSF.OS.Socket.
]
ProtocolSession                    # Etiqueta que especifica el protocolo que va a utilizar el enrutador.
[
  name tcp                          # En este caso para la capa de transporte se utiliza TCP.
  use SSF.OS.TCP.tcpSessionMaster # Significa que va a utilizar la clase tcpSessionMaster del
                                   #paquete SSF.OS.TCP.
  warn false                         # Desactiva el muestreo de las advertencias o Warnings.
]
ProtocolSession                    # Etiqueta que especifica el protocolo que va a utilizar el enrutador.
[
  name ip                            # En este caso para la capa de red se utiliza IP.
  use SSF.OS.IP                      # Significa que va a utilizar el paquete SSF.OS.IP.
]
]
interface                          # Instanciamiento de las interfaces.
[
  id 0                               # Identificador de las interfaces, en este caso la Intrefaz 0
                                   # correspondiente a la interfaz de loopback.
  virtual true                       # La interfaz de loopback se especifica como una interfaz virtual.
]
interface                          # Instanciamiento de las interfaces.
[
  idrange                            # Especifica el rango de interfaces que posee el enrutador.
  [
    from 1 to 4                      # En este caso las interfaces 1, 2 y 3.
  ]
]
nhi_route [ dest 2:2(0) interface 3 next_hop 2:1(3) ]
nhi_route [ dest 1:2(0) interface 4 next_hop 1:2(0) ]
]

host [                              # HTTP client
  id 2
  interface
  [
    id 0                             # El identificador del host es 0.
    bitrate 100000000                #La tasa de transferencia de bits es de 100000000 bytes
    latency 0.0                      #Latencia en los buffers de salida
  ]
  nhi_route                          # Dirección de la interfaz NHI relativa al router local que el cliente
                                   # debe usar cuando envíe los datos al enrutador BGP local.

  [
    dest 2:2(0)                      #Dirección del servidor http
    interface 0                       #interfaz de salida del host
    next_hop 1:1(4)                  #Próximo salto, a donde el cliente http enviará los datos
  ]
]
graph
[
  ProtocolSession
[

```

```

name client use SSF.OS.WWW.httpClient #tiempo mínimo en seg. de envío para la primera
                                     #petición al servidor http.

start_time 35.0                       # tiempo de inicio, 35 seg, la primera sesión de transferencia llegará
                                     #en el tiempo de inicio estipulado + el tiempo inter sesión en seg.

inter_session_time
[
distribution                           #distribución del No de páginas por sesión con un servidor.
[
name "Exponential"                     # nombre de la clase distribución
lambda 0.01
]
]
pages_in_session                       #distribución del tiempo de procesamiento del usuario (tiempos fuera)
                                     # Entre páginas consecutivas, contando desde el tiempo en seg. En
                                     #que se entrego completamente la página anterior.

[
distribution
[
name "Exponential"
lambda 0.2
]
]
inter_page_time
[
distribution
[
name "Pareto"                           #Nombre de la clase
k 25.0                                   # escala del parámetro
alpha 2.0                                # forma del parámetro (exponencial)
]
]
inter_request_time
[
distribution
[
name "Pareto"                           #Nombre de la clase
k 0.1667                                 # escala del parámetro
alpha 1.5                                # forma del parámetro (exponencial)
]
]
server_list "foobar"
_find .common.httpinit.http_hdr_size
_find .common.httpinit.show_report
_find .common.httpinit.persistent
_find .common.httpinit.debug
]
ProtocolSession
[ name socket use SSF.OS.Socket.socketMaster ]
ProtocolSession
[ name tcp use SSF.OS.TCP.tcpSessionMaster_find .common.tcpinit ]

```

```

ProtocolSession
  [name ip use SSF.OS.IP]
]
]
]

Net
  [
    id 4 # Se declara una red dentro de la topología.
    AS_status boundary # El identificador de la red es 4.
    router # Significa que esta red encierra un Sistema Autónomo.
      [
        id 1 # Especifica que en esta red existe un enrutador.
        graph # El identificador del enrutador es 1.
          # graph es un atributo que especifica una lista de protocolos que
          # pueden ser configurados dentro del enrutador y los nombres de las
          # clases del paquete SSF.OS que se van a usar.
        [
          ProtocolSession # Etiqueta que especifica el protocolo que va a utilizar el
enrutador.
            [
              name bgp
              use SSF.OS.BGP4.BGPSession # Significa que va a utilizar la clase BGPSession del paquete
# SSF.OS.BGP4.
              autoconfig false # Esto significa que la configuración automática esta desactivada,
# por lo cual deben especificarse todos los parámetros del enrutador.
              connretry_time 120 # Significa que deben transcurrir 120 seg. Antes de intentar reiniciar
# un intento fallido de conexión con un par BGP.
              min_as_orig_time 15 # Significa que la cantidad mínima de tiempo que debe pasar entre
# anuncios sucesivos e mensajes Update enviados.
              reflector false # La característica de route reflector esta desactivada.
              neighbor # Encapsula los atributos de un vecino BGP.
                [
                  as 5 # El identificador del AS vecino es 5.
                  address 1(1) # Dirección de la interfaz NHI relativa al AS vecino. En este caso
# 1(1) significa Enrutador 1(Interfaz 1).
                  use_return_address 1(2) # Dirección de la interfaz NHI relativa al AS local que el vecino
# debe usar cuando envíe mensajes al enrutador BGP local.
# En este caso debe usar la interfaz 2 del enrutador 1. Es decir
# Enrutador 1(Interfaz 2).
                  hold_time 90 # Tiempo máximo de espera en seg para que se pierda conectividad
# entre los pares BGP si no se recibe un mensaje Update o Keep Alive.
                  keep_alive_time 30 # Intervalo de tiempo en segundos para el envío de mensajes
# KeepAlive.
                  mrai 30 # Mínima cantidad de tiempo en segundos que debe transcurrir entre
# anuncios sucesivos de rutas.
                  infilter # Encapsula las políticas de filtrado para los anuncios de rutas
# provenientes desde este vecino.
                    [
                      clause # Permitir todas las rutas aprendidas desde el AS 5.
# Definición de una de las clausulas que determinará las políticas de
# filtrado.
                        [
                          precedence 1 # Especifica el orden en que serán evaluadas las clausulas cuando
# inicie la simulación, en este caso la número 1.

```

```

        predicate []      # Especifica un predicado asociado a la clausula correspondiente, el
                          # hecho de que este vacío significa que todas las rutas cumplen con
                          # Este predicado.

        action            # Acción que va a aplicarse a la ruta a través de esta clausula.
        [
        primary permit    # Acción primaria a ser aplicada en la ruta, esta puede ser permitir
                          # o denegar la acción.
        ]
    ]
]
outfilter                # Encapsula las políticas de filtrado para los anuncios de rutas con
                          # rumbo a este vecino.

[
_extends .filters.permit_all # Hereda del método filters ubicado en el archivo dictionary.dml y el
                              # cual permite todas las acciones sobre la ruta; es decir que no
                              # posee políticas de filtrado.
]
]
neighbor                # Encapsula los atributos de un vecino BGP.
[
  as 1                  # El identificador del AS vecino es 1.
  address 1(1)          # Dirección de la interfaz NHI relativa al AS vecino.
  use_return_address 1(1) # Dirección de la interfaz NHI relativa al AS local que el vecino
                          # debe usar cuando envíe mensajes al enrutador BGP local.
                          # En este caso debe usar la interfaz 1 del enrutador 1. Es decir
                          # Enrutador 1(Interfaz 1).

  hold_time 90          # Tiempo máximo de espera en seg para que se pierda conectividad
                          # entre los pares BGP si no se recibe un mensaje Update o Keep Alive.

  keep_alive_time 30    # Intervalo de tiempo en seg para el envío de mensajes KeepAlive.

  mrai 30               # Mínima cantidad de tiempo en segundos que debe transcurrir entre
                          # anuncios sucesivos de rutas.

  infilter              # Encapsula las políticas de filtrado para los anuncios de rutas
                          # provenientes desde este vecino.
  [
    clause              # Permitir todas las rutas aprendidas desde el AS 1.
                          # Definición de una de las clausulas que determinará las políticas de
                          # filtrado.
    [
      precedence 1      # Especifica el orden en que serán evaluadas las clausulas cuando
                          # inicie la simulación, en este caso la número 1.

      predicate []      # Especifica un predicado asociado a la clausula correspondiente, el
                          # hecho de que este vacío significa que todas las rutas cumplen con
                          # Este predicado.

      action            # Acción que va a aplicarse a la ruta a través de esta clausula.
      [
      primary permit    # Acción primaria a ser aplicada en la ruta, esta puede ser permitir
                          # o denegar la acción.
      ]
    ]
  ]
]
outfilter                # Encapsula las políticas de filtrado para los anuncios de rutas con
                          # rumbo a este vecino.

[
_extends .filters.permit_all # Hereda del método filters ubicado en el archivo dictionary.dml y el
                              # cual permite todas las acciones sobre la ruta; es decir que no
                              # posee políticas de filtrado.

```

```

    ]
  ]
neighbor                               # Encapsula los atributos de un vecino BGP.
[
  as 3                                  # El identificador del AS vecino es 3.
  address 1(1)                          # Dirección de la interfaz NHI relativa al AS vecino. En este caso
                                          # 1(1) significa Enrutador 1(Interfaz 1).
  use_return_address 1(3)               # Dirección de la interfaz NHI relativa al AS local que el vecino
                                          # debe usar cuando envíe mensajes al enrutador BGP local.
                                          # En este caso debe usar la interfaz 3 del enrutador 1. Es decir
                                          # Enrutador 1(Interfaz 3).
  hold_time 90                          # Tiempo máximo de espera en seg para que se pierda conectividad
                                          # entre los pares BGP si no se recibe un mensaje Update o Keep Alive.
  keep_alive_time 30                    # Intervalo de tiempo en seg para el envío de mensajes KeepAlive.
  mrai 30                                # Mínima cantidad de tiempo en segundos que debe transcurrir entre
                                          # anuncios sucesivos de rutas.
  infilter                               # Encapsula las políticas de filtrado para los anuncios de rutas
                                          # provenientes desde este vecino.
  [                                       # Permitir todas las rutas aprendidas desde el AS 3.
    clause
    [
      precedence 1                       # Especifica el orden en que serán evaluadas las clausulas cuando
                                          # inicie la simulación, en este caso la número 1.
      predicate []                       # Especifica un predicado asociado a la clausula correspondiente, el
                                          # hecho de que este vacío significa que todas las rutas cumplen con
                                          # Este predicado.
      action                              # Acción que va a aplicarse a la ruta a través de esta clausula.
      [
        primary permit                  # Acción primaria a ser aplicada en la ruta, esta puede ser permitir
                                          # o denegar la acción.
      ]
    ]
  ]
  outfilter                             # Encapsula las políticas de filtrado para los anuncios de rutas con
                                          # rumbo a este vecino.
  [
    _extends .filters.permit_all        # Hereda del método filters ubicado en el archivo dictionary.dml y el
                                          # cual permite todas las acciones sobre la ruta; es decir que no
                                          # posee políticas de filtrado.
  ]
]
ProtocolSession                         # Etiqueta que especifica el protocolo que va a utilizar el
                                          # enrutador.
[
  name socket                            # En este caso socket.
  use SSF.OS.Socket.socketMaster        # Significa que va a utilizar la clase socketMaster del paquete
                                          # SSF.OS.Socket
]
ProtocolSession                         # Etiqueta que especifica el protocolo que va a utilizar el enrutador.
[
  name tcp                               # En este caso para la capa de transporte se utiliza TCP.
  use SSF.OS.TCP.tcpSessionMaster      # Significa que va a utilizar la clase tcpSessionMaster del
                                          # paquete SSF.OS.TCP.
]

```



```

    warn false          # Desactiva el muestreo de las advertencias o Warnings.
  ]
  ProtocolSession     # Etiqueta que especifica el protocolo que va a utilizar el enrutador.
  [
    name ip            # En este caso para la capa de red se utiliza IP.
    use SSF.OS.IP      # Significa que va a utilizar el paquete SSF.OS.IP.
  ]
]
interface             # Instanciamiento de las interfaces.
  [
    id 0              # Identificador de las interfaces, en este caso la Intrefaz 0
                    # correspondiente a la interfaz de loopback.
    virtual true      # La interfaz de loopback se especifica como una interfaz virtual.
  ]
interface             # Instanciamiento de las interfaces.
  [
    idrange           # Especifica el rango de interfaces que posee el enrutador.
    [
      from 1 to 3     # En este caso las interfaces 1, 2 y 3.
    ]
  ]
]
]

Net
[
  id 3                # Se declara una red dentro de la topología.
  AS_status boundary # El identificador de la red es 3.
                    # Significa que esta red encierra un Sistema Autónomo, o lo que es lo
                    # mismo que esta red es el AS 3.
  router              # Especifica que en esta red existe un enrutador.
  [
    id 1              # El identificador del enrutador es 1.
    graph             # graph es un atributo que especifica una lista de protocolos que
                    # pueden ser configurados dentro del enrutador y los nombres de las
                    # clases del paquete SSF.OS que se van a usar.
  ]
  [
    ProtocolSession  # Etiqueta que especifica el protocolo que va a utilizar el enrutador.
    [
      name bgp
      use SSF.OS.BGP4.BGPSession # Significa que va a utilizar la clase BGPSession del paquete
                                # SSF.OS.BGP4.
      autoconfig false # Esto significa que la configuración automática esta desactivada,
                       # por lo cual deben especificarse todos los parámetros del enrutador.
      connretry_time 120 # Significa que deben transcurrir 120 seg. antes de intentar reiniciar
                        # un intento fallido de conexión con un par BGP.
      min_as_orig_time 15 # Significa que la cantidad mínima de tiempo que debe pasar entre
                          # anuncios sucesivos y mensajes Update enviados.
      reflector false    # La característica de route reflector esta desactivada.
      neighbor           # Encapsula los atributos de un vecino BGP.
      [
        as 5            # El identificador del AS vecino es 5.
        address 1(4)    # Dirección de la interfaz NHI relativa al AS vecino. En este caso
                        # 1(4) significa Enrutador 1(Interfaz 4).
      ]
    ]
  ]
]

```

```

use_return_address 1(2)    # Dirección de la interfaz NHI relativa al AS local que el vecino
                          # debe usar cuando envíe mensajes al enrutador BGP local.
                          # En este caso debe usar la interfaz 2 del enrutador 1. Es decir
                          # Enrutador 1(Interfaz 2).
_extends .basic_ebgp_neighbor # Significa que hereda los otros parámetros para el enrutador del
                              # metodo basic_ebgp_neighbor presente en el archivo dictionary.dml
    ]
neighbor
    [
        as 4                # El identificador del AS vecino es 4.
        address 1(3)        # Dirección de la interfaz NHI relativa al AS vecino. En este caso
                          # 1(3) significa Enrutador 1(Interfaz 3).
        use_return_address 1(1) # Dirección de la interfaz NHI relativa al AS local que el vecino
                          # debe usar cuando envíe mensajes al enrutador BGP local.
                          # En este caso debe usar la interfaz 1 del enrutador 1. Es decir
                          # Enrutador 1(Interfaz 1).
        _extends .basic_ebgp_neighbor # Significa que hereda los otros parámetros para el enrutador del
                              # metodo basic_ebgp_neighbor presente en el archivo dictionary.dml
    ]
neighbor
    [
        as 2                # El identificador del AS vecino es 2.
        address 1(1)        # Dirección de la interfaz NHI relativa al AS vecino. En este caso
                          # 1(1) significa Enrutador 1(Interfaz 1).
        use_return_address 1(3) # Dirección de la interfaz NHI relativa al AS local que el vecino
                          # debe usar cuando envíe mensajes al enrutador BGP local.
                          # En este caso debe usar la interfaz 3 del enrutador 1. Es decir
                          # Enrutador 1(Interfaz 3).
        _extends .basic_ebgp_neighbor # Significa que hereda los otros parámetros para el enrutador del
                              # metodo basic_ebgp_neighbor presente en el archivo dictionary.dml
    ]
]
ProtocolSession            # Etiqueta que especifica el protocolo que va a utilizar el enrutador.
    [
        name socket        # En este caso socket.
        use SSF.OS.Socket.socketMaster # Significa que va a utilizar la clase socketMaster del
                                      # paquete SSF.OS.Socket.
    ]
ProtocolSession            # Etiqueta que especifica el protocolo que va a utilizar el enrutador.
    [
        name tcp           # En este caso para la capa de transporte se utiliza TCP.
        use SSF.OS.TCP.tcpSessionMaster # Significa que va a utilizar la clase tcpSessionMaster del
                                      # paquete SSF.OS.TCP.
        warn false        # Desactiva el muestreo de las advertencias o Warnings.
    ]
ProtocolSession            # Etiqueta que especifica el protocolo que va a utilizar el enrutador.
    [
        name ip            # En este caso para la capa de red se utiliza IP.
        use SSF.OS.IP      # Significa que va a utilizar el paquete SSF.OS.IP.
    ]
]
interface                  # Instanciamiento de las interfaces.
    [

```

```

    id 0                                # Identificador de las interfaces, en este caso la Interfaz 0
    virtual true                         # correspondiente a la interfaz de loopback.
    ]                                     # La interfaz de loopback se especifica como una interfaz virtual.
  interface                             # Instanciamiento de las interfaces.
  [
    idrange                              # Especifica el rango de interfaces que posee el enrutador.
    [
      from 1 to 3                        # En este caso las interfaces 1, 2 y 3.
    ]
  ]
]
]

Net
[
  id 2                                  # Se declara una red dentro de la topología.
  AS_status boundary                   # El identificador de la red es 2.
  router                               # Significa que esta red encierra un Sistema Autónomo, o lo que es lo
  [                                     # mismo que esta red es el AS 2.
    id 1                                # Especifica que en esta red existe un enrutador.
    graph                               # El identificador del enrutador es 1.
    [                                   # graph es un atributo que especifica una lista de protocolos que
      ProtocolSession                   # pueden ser configurados dentro del enrutador y los nombres de las
      [                                   # clases del paquete SSF.OS que se van a usar.
        name bgp                         # Etiqueta que especifica el protocolo que va a utilizar el enrutador.
        use SSF.OS.BGP4.BGPSession      # Significa que va a utilizar la clase BGPSession del paquete
        autoconfig false                 # SSF.OS.BGP4.
        connretry_time 120               # Esto significa que la configuracion automatoca esta desactivada,
        min_as_orig_time 15              # por lo cual deben especificarse todos los parametros del enrutador.
        reflector false                  # Significa que deben transcurrir 120 seg. antes de intentar reiniciar
        neighbor                         # un intento fallido de conexión con un par BGP.
        [
          as 5                           # Significa que la cantidad mínima de tiempo que debe pasar entre
          address 1(3)                    # anuncios sucesivos y mensajes Update.
          use_return_address 1(2)         # La característica de route reflector esta desactivada.
          _extends .basic_ebgp_neighbor  # Encapsula los atributos de un vecino BGP.
        ]
      ]
    ]
  ]
  neighbor
  [
    as 5                                # El identificador del AS vecino es 5.
    address 1(3)                         # Dirección de la interfaz NHI relativa al AS vecino. En este caso
    use_return_address 1(2)              # 1(3) significa Enrutador 1(Interfaz 3).
    _extends .basic_ebgp_neighbor        # Dirección de la interfaz NHI relativa al AS local que el vecino
    ]                                     # debe usar cuando envíe mensajes al enrutador BGP local.
  ]                                     # En este caso debe usar la interfaz 2 del enrutador 1. Es decir
  neighbor                               # Enrutador 1(Interfaz 2).
  [
    as 3                                # Significa que hereda los otros parámetros para el enrutador del
    ]                                     # método basic_ebgp_neighbor presente en el archivo dictionary.dml
  ]
]
]

```

```

        address 1(3)          # Dirección de la interfaz NHI relativa al AS vecino. En este caso
                              # 1(3) significa Enrutador 1(Interfaz 3).
        use_return_address 1(1) # Dirección de la interfaz NHI relativa al AS local que el vecino
                              # debe usar cuando envíe mensajes al enrutador BGP local.
                              # En este caso debe usar la interfaz 1 del enrutador 1. Es decir
                              # Enrutador 1(Interfaz 1).
        _extends .basic_ebgp_neighbor # Significa que hereda los otros parámetros para el enrutador del
                                      # método basic_ebgp_neighbor presente en el archivo dictionary.dml
    ]
neighbor
    [
        as 1                    # El identificador del AS vecino es 1.
        address 1(3)           # Dirección de la interfaz NHI relativa al AS vecino. En este caso
                              # 1(3) significa Enrutador 1(Interfaz 3).
        use_return_address 1(3) # Dirección de la interfaz NHI relativa al AS local que el vecino
                              # debe usar cuando envíe mensajes al enrutador BGP local.
                              # En este caso debe usar la interfaz 3 del enrutador 1. Es decir
                              # Enrutador 1(Interfaz 3).
        _extends .basic_ebgp_neighbor # Significa que hereda los otros parámetros para el enrutador del
                                      # método basic_ebgp_neighbor presente en el archivo dictionary.dml
    ]
]
ProtocolSession                # Etiqueta que especifica el protocolo que va a utilizar el enrutador.
    [
        name socket            # En este caso socket.
        use SSF.OS.Socket.socketMaster # Significa que va a utilizar la clase socketMaster del
                                      # paquete SSF.OS.Socket.
    ]
ProtocolSession                # Etiqueta que especifica el protocolo que va a utilizar el enrutador.
    [
        name tcp               # En este caso para la capa de transporte se utiliza TCP.
        use SSF.OS.TCP.tcpSessionMaster # Significa que va a utilizar la clase tcpSessionMaster
                                      # del paquete SSF.OS.TCP.
        warn false            # Desactiva el muestreo de las advertencias o Warnings.
    ]
ProtocolSession                # Etiqueta que especifica el protocolo que va a utilizar el enrutador.
    [
        name ip                # En este caso para la capa de red se utiliza IP.
        use SSF.OS.IP          # Significa que va a utilizar el paquete SSF.OS.IP.
    ]
]
Interface                       # Instanciamiento de las interfaces.
    [
        id 0                   # Identificador de las interfaces, en este caso la Interfaz 0
                              # correspondiente a la interfaz de loopback.
        virtual true           # La interfaz de loopback se especifica como una interfaz virtual.
    ]
interface                       # Instanciamiento de las interfaces.
    [
        idrange                # Especifica el rango de interfaces que posee el enrutador.
        [
            from 1 to 4        # En este caso las interfaces 1, 2 y 3.
        ]
    ]

```

```

]
nhi_route [ dest 2:2(0) interface 4 next_hop 2:2(0) ]
nhi_route [ dest 1:2(0) interface 3 next_hop 1:1(3) ]
]

# HTTP server
host [
  id 2
  interface
    [
      id 0                # El identificador del host es 0.
      bitrate 100000000   #La tasa de transferencia de bits es de 100000000 bytes
      latency 0.0         #Latencia en los buffers de salida
    ]
    nhi_route            # Dirección de la interfaz NHI relativa al router local que el cliente
    [
      dest 1:2(0)        #Dirección del servidor http
      interface 0        #interfaz de salida del host
      next_hop 2:1(4)    #Próximo salto, a donde el cliente http enviará los datos
    ]
  ]
  graph
  [
    ProtocolSession
    [
      name server use SSF.OS.WWW.httpServer    #tiempo mínimo en seg. de envío para la
                                                # primera petición al servidor http.

      port 80                # puerto del servidor HTTP
      client_limit 10        # número máximo de clientes permitidos
      objects_in_page
      [
        distribution
        [
          name "Pareto"      #Nombre de la clase
          k 0.6667           # escala del parámetro
          alpha 1.2         # forma del parámetro (exponencial)
        ]
      ]
    ]
  ]
  object_size
  [
    distribution
    [
      name "Pareto"        #Nombre de la clase
      k 0.6667             # escala del parámetro
      alpha 1.2           # forma del parámetro (exponencial)
    ]
  ]
]

response_delay
[
  distribution
  [

```

```

    name "Exponential"
    lambda 10.0
  ]
]

_find .common.httppinit.http_hdr_size
_find .common.httppinit.show_report
_find .common.httppinit.debug
]
ProtocolSession [name socket use SSF.OS.Socket.socketMaster]
ProtocolSession [name tcp use SSF.OS.TCP.tcpSessionMaster
    _find .common.tcppinit]
ProtocolSession [name ip use SSF.OS.IP]
]
]
]

link # Atributo que especifica la interconexión de las interfaces de los
      # enrutadores a nivel 2.
[
attach 5:1(1) # Especifica que se enlaza la interfaz 1 del enrutador 1 perteneciente
              # al AS 5, así AS 5:Enrutador 1 (Interfaz 1).
attach 4:1(2) # con la interfaz 2 del enrutador 1 perteneciente al AS 4, así
              # AS 4: Enrutador 1 (Interfaz 2).
]
link # Atributo que especifica la interconexión de las interfaces de los
      # enrutadores a nivel 2.
[
attach 5:1(2) # Especifica que se enlaza la interfaz 2 del enrutador 1 perteneciente
              # al AS 5, así AS 5:Enrutador 1 (Interfaz 2).
attach 1:1(2) # con la interfaz 2 del enrutador 1 perteneciente al AS 1, así
              # AS 1: Enrutador 1 (Interfaz 2).
]
link # Atributo que especifica la interconexión de las interfaces de los
      # enrutadores a nivel 2.
[
attach 5:1(3) # Especifica que se enlaza la interfaz 3 del enrutador 1 perteneciente
              # al AS 5, así AS 5:Enrutador 1 (Interfaz 3).
attach 2:1(2) # con la interfaz 2 del enrutador 1 perteneciente al AS 2, así
              # AS 2: Enrutador 1 (Interfaz 2).
]
link # Atributo que especifica la interconexión de las interfaces de los
      # enrutadores a nivel 2.
[
attach 5:1(4) # Especifica que se enlaza la interfaz 4 del enrutador 1 perteneciente
              # al AS 5, así AS 5:Enrutador 1 (Interfaz 4).
attach 3:1(2) # con la interfaz 2 del enrutador 1 perteneciente al AS 3, así
              # AS 3: Enrutador 1 (Interfaz 2).
]
link # Atributo que especifica la interconexión de las interfaces de los
      # enrutadores a nivel 2.
[

```

```

attach 1:1(1)          # Especifica que se enlaza la interfaz 1 del enrutador 1 perteneciente
                      # al AS 1, así AS 1:Enrutador 1 (Interfaz 1).
attach 4:1(1)          # con la interfaz 1 del enrutador 1 perteneciente al AS 4, así
                      # AS 4: Enrutador 1 (Interfaz 1).
]                      # Retardo asociado al enlace de 0.01 segs.
link                  # Atributo que especifica la interconexión de las interfaces de los
                      # enrutadores a nivel 2.
[
attach 1:1(3)          # Especifica que se enlaza la interfaz 3 del enrutador 1 perteneciente
                      # al AS 1, así AS 1:Enrutador 1 (Interfaz 3).
attach 2:1(3)          # con la interfaz 3 del enrutador 1 perteneciente al AS 2, así
                      # AS 2: Enrutador 1 (Interfaz 3).
]
link                  # Atributo que especifica la interconexión de las interfaces de los
                      # enrutadores a nivel 2.
[
attach 2:1(1)          # Especifica que se enlaza la interfaz 1 del enrutador 1 perteneciente
                      # al AS 2, así AS 2:Enrutador 1 (Interfaz 1).
attach 3:1(3)          # con la interfaz 3 del enrutador 1 perteneciente al AS 3, así
                      # AS 3: Enrutador 1 (Interfaz 3).
]
link                  # Atributo que especifica la interconexión de las interfaces de los
                      # enrutadores a nivel 2.
[
attach 3:1(1)          # Especifica que se enlaza la interfaz 1 del enrutador 1 perteneciente
                      # al AS 3, así AS 3:Enrutador 1 (Interfaz 1).
attach 4:1(3)          # con la interfaz 3 del enrutador 1 perteneciente al AS 4, así
                      # AS 4: Enrutador 1 (Interfaz 3).
]
link
[
attach 1:1(4)          # Especifica que se enlaza la interfaz 4 del enrutador 1 perteneciente
                      # al AS 1, así AS 1:Enrutador 1 (Interfaz 4).
attach 1:2(0)          # con la interfaz 0 del enrutador 2 perteneciente al AS 1, así
                      # AS 1: Enrutador 2 (Interfaz 0).
]
link
[
attach 2:1(4)          # Especifica que se enlaza la interfaz 4 del enrutador 1 perteneciente
                      # al AS 2, así AS 2:Enrutador 1 (Interfaz 4).
attach 2:2(0)          # con la interfaz 0 del enrutador 2 perteneciente al AS 2, así
                      # AS 2: Enrutador 2 (Interfaz 0).
]
]

common
[
tcpinit
[
ISS 10000              # Número inicial de secuencia
MSS 2000              # tamaño máximo del segmento
RcvWndSize 32         # tamaño del buffer receptor
SendWndSize 32        # tamaño máximo de la ventana de transmisión

```

```

SendBufferSize 128          # tamaño del buffer transmisor
MaxRexmitTimes 12          # número máximo de retransmisiones antes de descartar
TCP_SLOW_INTERVAL 0.5     # granularidad TCP de tiempo largo
TCP_FAST_INTERVAL 0.2     # granularidad TCP de tiempo corto, retardo ACK
MSL 60.0                  # tiempo de vida máximo del segmento
MaxIdleTime 600.0         # tiempo de espera máximo antes de terminar la conexión.
delayed_ack false         # opción ACK retardado
fast_recovery true        # algoritmo de implementación
show_report false         # imprime un resumen del reporte de conexión.
debug false               # imprime un reporte completo
]
httpinit
[
  http_hdr_size 1000      # tamaño del encabezado http (bytes virtuales) leído de un socket.
                          # Antes de leer los datos (si es indicado por el encabezado http)
                          # es falso, si es verdadero, una conexión insistente TCP por sesión.
                          # imprime un resumen del reporte de la sesión cliente servidor.
                          # imprime un reporte completo del diagnostico entre cliente servidor
  show_report true
  debug false
]
]

```

4.2. ESCENARIO 2

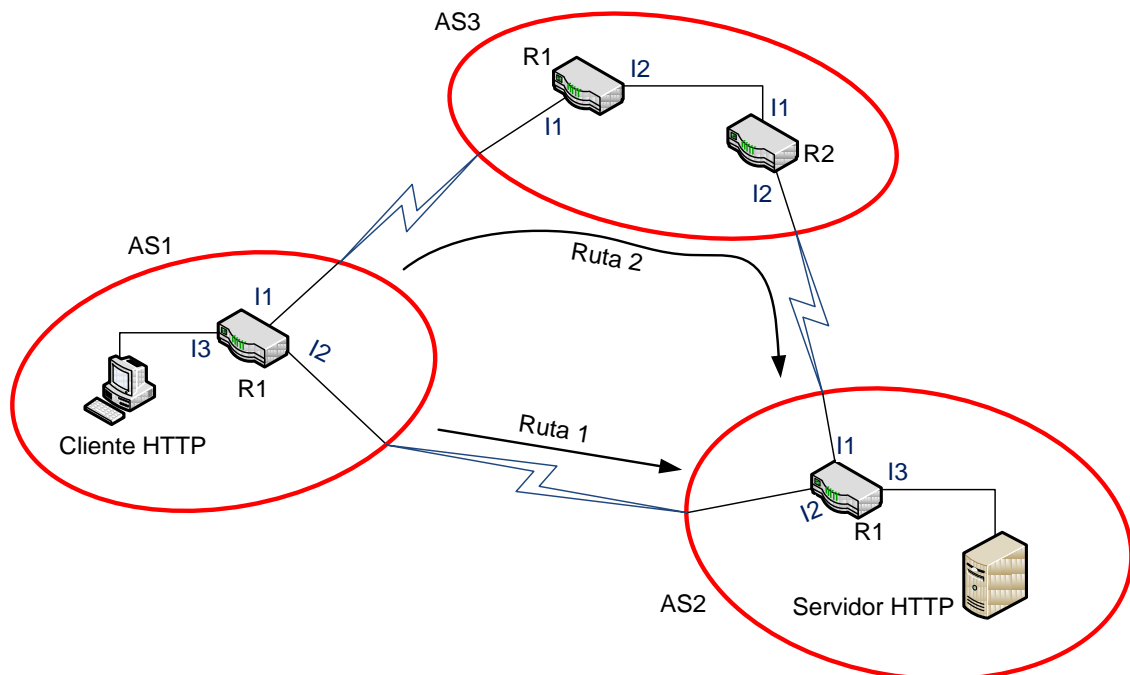


Figura 6 Escenario de simulación 2.


```
_schema
[
  _find .schemas.Net
]

Net
[
  frequency 1000000000
  randomstream
  [
    generator "MersenneTwister"
    stream "seedstarter2"
    reproducibility_level "host"
  ]
]

traffic
[
  pattern
  [
    client 1:2
    servers
    [
      list "foobar" nhi 2:2(0) port 80
    ]
  ]
]

bgpoptions
[
  validation_test 12
  startup_jitter_bound 0.1
]

Net
[
  id 1
  AS_status boundary
  router
  [
    id 1
    graph
    [
      ProtocolSession
      [
        name bgp
        use SSF.OS.BGP4.BGPSSession
        autoconfig false
        connretry_time 120
        min_as_orig_time 15
        reflector false
      ]
    ]
  ]
]
```

```
neighbor
  [
  as 3
  address 1(1)
  use_return_address 1(1)
  hold_time 90
  keep_alive_time 30
  mrai 30
  infilter
    [
    clause
      [
      precedence 1
      predicate []
      action
        [
        primary permit
        ]
      ]
    ]
  outfilter
    [
    _extends .filters.permit_all
    ]
  ]
neighbor
  [
  as 2
  address 1(2)
  use_return_address 1(2)
  hold_time 90
  keep_alive_time 30
  mrai 30
  infilter.
    [
    clause
      [
      precedence 1
      predicate []
      action
        [
        primary permit
        ]
      ]
    ]
  outfilter
    [
    _extends .filters.permit_all
    ]
  ]
]
```

```
ProtocolSession
[
  name socket
  use SSF.OS.Socket.socketMaster
]
ProtocolSession
[
  name tcp
  use SSF.OS.TCP.tcpSessionMaster
  warn false
]
ProtocolSession
[
  name ip
  use SSF.OS.IP
]
]
interface
[
  id 0
  virtual true
]
interface
[
  idrange
  [
    from 1 to 3
  ]
]
]

# Cliente HTTP
host
[
  id 2
  interface
  [
  id 0
  bitrate 100000000
  latency 0.0
  ]
  nhi_route
  [
  dest 2:2(0)
  interface 0
  next_hop 1:1(3)
  ]
  graph
  [
    ProtocolSession
  ]
]
```

```
name client
use SSF.OS.WWW.httpClient
  start_time 35.0
  inter_session_time
  [
  distribution
  [
  name "Exponential"
  lambda 0.01
  ]
  ]
  pages_in_session
  [
  distribution
  [
  name "Exponential"
  lambda 0.2
  ]
  ]
  inter_page_time
  [
  distribution
  [
  name "Pareto"
  k 25.0
  alpha 2.0
  ]
  ]
  inter_request_time
  [
  distribution
  [
  name "Pareto"
  k 0.1667
  alpha 1.5
  ]
  ]
  server_list "foobar"
    _find .common.httppinit.http_hdr_size
    _find .common.httppinit.show_report
    _find .common.httppinit.persistent
    _find .common.httppinit.debug
  ]
  ProtocolSession
  [
  name socket
  use SSF.OS.Socket.socketMaster
  ]
  ProtocolSession
  [
  name tcp
```

```
    use SSF.OS.TCP.tcpSessionMaster_find .common.tcpinit
  ]
  ProtocolSession
  [
    name ip use SSF.OS.IP
  ]
]
]
]

Net
[
  id 3
  AS_status boundary
  router
  [
    id 1
    graph
    [
      ProtocolSession
      [
        name bgp
        use SSF.OS.BGP4.BGPSession
        autoconfig false
        connretry_time 120
        min_as_orig_time 15
        reflector false
        neighbor
        [
          as 1
          address 1(1)
          use_return_address 1(1)
          _extends .basic_ebgp_neighbor
        ]
        neighbor
        [
          as 3
          address 2(1)
          use_return_address 1(2)
          _extends .basic_ebgp_neighbor
        ]
      ]
      ProtocolSession
      [
        name socket
        use SSF.OS.Socket.socketMaster
      ]
      ProtocolSession
      [
        name tcp
        use SSF.OS.TCP.tcpSessionMaster
```

```
warn false
]
ProtocolSession
[
  name ip
  use SSF.OS.IP
]
]
interface
[
  id 0
  virtual true
]
interface
[
  idrange
  [
    from 1 to 2
  ]
]
]
]

router
[
  id 2
  graph
  [
    ProtocolSession
    [
      name bgp
      use SSF.OS.BGP4.BGP4Session
      autoconfig false
      connretry_time 120
      min_as_orig_time 15
      reflector false
      neighbor
      [
        as 2
        address 1(1)
        use_return_address 2(2)
        _extends .basic_ebgp_neighbor
      ]
      neighbor
      [
        as 3
        address 1(2)
        use_return_address 2(1)
        _extends .basic_ebgp_neighbor
      ]
    ]
  ]
]
ProtocolSession
```

```
[
  name socket
  use SSF.OS.Socket.socketMaster
]
ProtocolSession
[
  name tcp
  use SSF.OS.TCP.tcpSessionMaster
  warn false
]
ProtocolSession
[
  name ip
]
]
interface
  [
    id 0
    virtual true
  ]
interface
  [
    idrange
    [
      from 1 to 2
    ]
  ]
]
]
]

Net
[
  id 2
  AS_status boundary
  router
  [
    id 1
    graph
    [
      ProtocolSession
      [
        name bgp
        use SSF.OS.BGP4.BGPSession
        autoconfig false
        connretry_time 120
        min_as_orig_time 15
        reflector false
        neighbor
        [
          as 3
          address 2(2)
        ]
      ]
    ]
  ]
]
]
```

```
        use_return_address 1(1)
        _extends .basic_ebgp_neighbor
    ]
neighbor
    [
        as 1
        address 1(2)
        use_return_address 1(2)
        _extends .basic_ebgp_neighbor
    ]
]
ProtocolSession
[
    name socket
    use SSF.OS.Socket.socketMaster
]
ProtocolSession
[
    name tcp
    use SSF.OS.TCP.tcpSessionMaster
    warn false
]
ProtocolSession
[
    name ip
    use SSF.OS.IP
]
]
interface
[
    id 0
    virtual true
]
interface
[
    idrange
    [
        from 1 to 3
    ]
]
]
]

# Servidor HTTP
host
[
    id 2
    interface
    [
        id 0
        bitrate 100000000
        latency 0.0
    ]
]
]
```



```
]
nhi_route
[
dest 1:2(0)
interface 0
next_hop 2:1(3)
]
graph
[
  ProtocolSession
  [
name server
use SSF.OS.WWW.httpServer
port 80
client_limit 10
objects_in_page
[
distribution
[
name "Pareto"
k 0.6667
alpha 1.2
]
]
object_size
[
distribution
[
name "Pareto"
k 2000.0
alpha 1.2
]
]
response_delay
[
distribution
[
name "Exponential"
lambda 10.0
]
]
]
_find .common.httpinit.http_hdr_size
_find .common.httpinit.show_report
_find .common.httpinit.debug
]
ProtocolSession
[
name socket
use SSF.OS.Socket.socketMaster
]
ProtocolSession
```

```
[
  name tcp
  use SSF.OS.TCP.tcpSessionMaster_find .common.tcpinit
]
ProtocolSession
[
  name ip
  use SSF.OS.IP
]
]
]
]

link
[
  attach 1:1(1)
  attach 3:1(1)
]
link
[
  attach 3:1(2)
  attach 3:2(1)
]
link
[
  attach 3:2(2)
  attach 2:1(1)
]
link
[
  attach 2:1(3)
  attach 2:2(0)
]
link
[
  attach 2:1(2)
  attach 1:1(2)
]
link.
[
  attach 1:1(3)
  attach 1:2(0)
]
]

common
[
  tcpinit
  [
    ISS 10000
    MSS 2000
  ]
]
```

```

RcvWndSize 32
SendWndSize 32
SendBufferSize 128
MaxRexmitTimes 12
TCP_SLOW_INTERVAL 0.5
TCP_FAST_INTERVAL 0.2
MSL 60.0
MaxIdleTime 600.0
delayed_ack false
fast_recovery true
show_report false
debug false
]
httpinit
[
  http_hdr_size 1000
  persistent false
  show_report true
  debug false
]
]

```

4.3. ESCENARIO 3

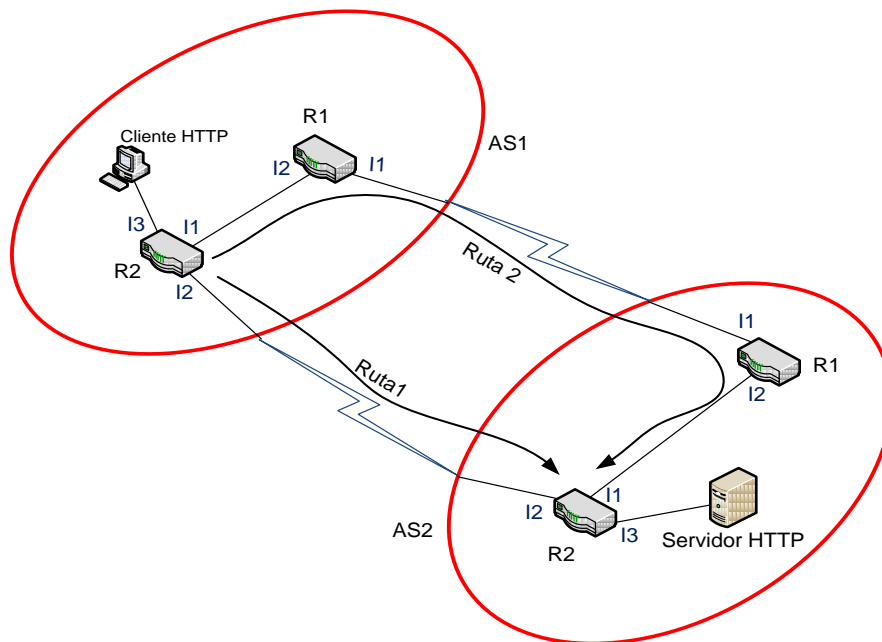


Figura 7 Escenario de simulación 3

```
_schema
[
  _find .schemas.Net
]

Net
[
  frequency 1000000000

  randomstream
  [
    generator "MersenneTwister"
    stream "seedstarter2"
    reproducibility_level "host"
  ]

  traffic
  [
    pattern
    [
      client 1:3
      servers
      [
        list "foobar" nhi 2:3(0) port 80
      ]
    ]
  ]

  bgpoptions
  [
    validation_test 12
    startup_jitter_bound 0.1
  ]

  Net
  [
    id 1
    AS_status boundary
    router
    [
      id 1
      graph
      [
        ProtocolSession
        [
          name bgp
          use SSF.OS.BGP4.BGPSession
          autoconfig false
          connretry_time 120
          min_as_orig_time 15
          reflector false
        ]
      ]
    ]
  ]
]
```

```
neighbor
[
  as 2
  address 1(1)
  use_return_address 1(1)
  hold_time 90
  keep_alive_time 30
  mrai 30
  infilter
  [
    clause
    [
      precedence 1.
      predicate []
      action
      [
        primary permit
      ]
    ]
  ]
  outfilter
  [
    _extends .filters.permit_all
  ]
]
neighbor
[
  as 1
  address 2(1)
  use_return_address 1(2)
  hold_time 90
  keep_alive_time 30
  mrai 30.
  infilter
  [
    clause
    [
      precedence 1
      predicate []
      action
      [
        primary permit
      ]
    ]
  ]
  outfilter
  [
    _extends .filters.permit_all
  ]
]
]
```

```
ProtocolSession
[
name socket
use SSF.OS.Socket.socketMaster
]
ProtocolSession
[
name tcp
use SSF.OS.TCP.tcpSessionMaster
warn false
]
ProtocolSession
[
name ip
use SSF.OS.IP
]
]
interface
[
id 0
virtual true
]
interface
[
idrange
[
from 1 to 2
]
]
]
router
[
id 2
graph
[
ProtocolSession
[
name bgp
use SSF.OS.BGP4.BGPSession
autoconfig false
connretry_time 120
min_as_orig_time 15
reflector false
neighbor
[
as 2
address 2(2)
use_return_address 2(2)
hold_time 90
keep_alive_time 30
```

```
mrai 30
infilter
[
  clause
  [
    precedence 1
    predicate []
    action
    [
      primary permit
    ]
  ]
]
outfilter
[
  _extends .filters.permit_all
]
]
neighbor
[
  as 1
  address 1(2)
  use_return_address 2(1)
  hold_time 90
  keep_alive_time 30
  mrai 30
  infilter
  [
    clause
    [
      precedence 1
      predicate []
      action
      [
        primary permit
      ]
    ]
  ]
  outfilter
  [
    _extends .filters.permit_all
  ]
]
]
ProtocolSession
[
  name socket
  use SSF.OS.Socket.socketMaster
]
ProtocolSession
[
```

```
name tcp
use SSF.OS.TCP.tcpSessionMaster
warn false
]
ProtocolSession
[
name ip
use SSF.OS.IP
]
]
interface
[
id 0
virtual true
]
interface
[
idrange
[
from 1 to 3
]
]
]

# Cliente HTTP
host
[
id 3
interface
[
id 0
bitrate 100000000
latency 0.0
]
nhi_route
[
dest 2:3(0)
interface 0
next_hop 1:2(3)
]
graph
[
ProtocolSession
[
name client
use SSF.OS.WWW.httpClient
start_time 35.0
inter_session_time
[
distribution
[
```



```
    name "Exponential"
    lambda 0.01
  ]
]
pages_in_session
[
  distribution
  [
    name "Exponential"
    lambda 0.2
  ]
]
inter_page_time
[
  distribution
  [
    name "Pareto"
    k 25.0
    alpha 2.0
  ]
]
inter_request_time
[
  distribution
  [
    name "Pareto"
    k 0.1667
    alpha 1.5
  ]
]
server_list "foobar"
_find .common.httppinit.http_hdr_size
_find .common.httppinit.show_report
_find .common.httppinit.persistent
_find .common.httppinit.debug
]
ProtocolSession
[
  name socket
  use SSF.OS.Socket.socketMaster
]
ProtocolSession
[
  name tcp
  use SSF.OS.TCP.tcpSessionMaster_find .common.tcpinit
]
ProtocolSession
[
  name ip
  use SSF.OS.IP
]
```

```
    ]
  ]
]

Net
[
  id 2
  AS_status boundary
  router
  [
    id 1
    graph
    [
      ProtocolSession
      [
        name bgp
        use SSF.OS.BGP4.BGPSession
        autoconfig false
        connretry_time 120
        min_as_orig_time 15
        reflector false
        neighbor
        [
          as 1
          address 1(1)
          use_return_address 1(1)
          hold_time 90
          keep_alive_time 30
          mrai 30
          infilter
          [
            clause
            [
              precedence 1
              predicate []
              action
              [
                primary permit
              ]
            ]
          ]
          outfilter
          [
            _extends .filters.permit_all
          ]
        ]
      ]
      neighbor
      [
        as 2
        address 2(1)
        use_return_address 1(2)
      ]
    ]
  ]
]
```

```
        _extends .basic_ebgp_neighbor
    ]
]
ProtocolSession
[
name socket
use SSF.OS.Socket.socketMaster
]
ProtocolSession
[
name tcp
use SSF.OS.TCP.tcpSessionMaster
warn false
]
ProtocolSession
[
name ip
use SSF.OS.IP
]
]
interface
[
id 0
virtual true
]
interface
[
idrange
[
from 1 to 2
]
]
]

router
[
id 2
graph
[
ProtocolSession
[
name bgp
use SSF.OS.BGP4.BGPSession
autoconfig false
connretry_time 120
min_as_orig_time 15
reflector false
neighbor
[
as 1
address 2(2
```

```
    use_return_address 2(2)
    hold_time 90
    keep_alive_time 30
    mrai 30
    infilter
    [
    clause
    [
    precedence 1
    predicate []
    action
    [
    primary permit
    ]
    ]
    ]
    outfilter
    [
    _extends .filters.permit_all
    ]
    ]
    neighbor
    [
    as 2
    address 1(2)
    use_return_address 2(1)
    _extends .basic_ebgp_neighbor
    ]
    ]
    ProtocolSession
    [
    name socket
    use SSF.OS.Socket.socketMaster
    ]
    ProtocolSession
    [
    name tcp
    use SSF.OS.TCP.tcpSessionMaster
    warn false
    ]
    ProtocolSession
    [
    name ip
    use SSF.OS.IP
    ]
    ]
    interface
    [
    id 0
    virtual true
    ]
```

```
interface
  [
    idrange
    [
      from 1 to 3
    ]
  ]
]

# Servidor HTTP
host
[
id 3
interface
  [
    id 0
    bitrate 100000000
    latency 0.0
  ]
  nhi_route
  [
    dest 1:3(0)
    interface 0
    next_hop 2:2(3)
  ]
  graph
  [
    ProtocolSession
    [
      name server
      use SSF.OS.WWW.httpServer
      port 80
      client_limit 10
      objects_in_page
      [
        distribution
        [
          name "Pareto"
          k 0.6667
          alpha 1.2
        ]
      ]
    ]
    object_size
    [
      distribution
      [
        name "Pareto"
        k 2000.0
        alpha 1.2
      ]
    ]
  ]
]
```

```

    response_delay
    [
    distribution
    [
    name "Exponential"
    lambda 10.0
    ]
    ]
_find .common.httpinit.http_hdr_size
_find .common.httpinit.show_report
_find .common.httpinit.debug
]
  ProtocolSession
  [
  name socket
  use SSF.OS.Socket.socketMaster
  ]
  ProtocolSession
  [
  name tcp
  use SSF.OS.TCP.tcpSessionMaster_find .common.tcpinit
  ]
  ProtocolSession
  [
  name ip
  use SSF.OS.IP
  ]
  ]
  ]
]

link
[
attach 1:1(1)
attach 2:1(1)
]
link
[
attach 2:1(2)
attach 2:2(1)
]
link
[
attach 2:2(2)
attach 1:2(2)
]
link.
[
attach 1:2(1)
attach 1:1(2)
]

```

```
link
[
  attach 1:2(3)
  attach 1:3(0)
]
link
[
  attach 2:2(3)
  attach 2:3(0)
]
]

common
[

  tcpinit

  [
    ISS 10000
    MSS 2000
    RcvWndSize 32
    SendWndSize 32
    SendBufferSize 128
    MaxRexmitTimes 12
    TCP_SLOW_INTERVAL 0.5
    TCP_FAST_INTERVAL 0.2
    MSL 60.0
    MaxIdleTime 600.0
    delayed_ack false
    fast_recovery true
    show_report false
    debug false
  ]

  httpinit
  [
    http_hdr_size 1000
    persistent false
    show_report true
    debug false
  ]
]
]
```

4.4. ESCENARIO 4

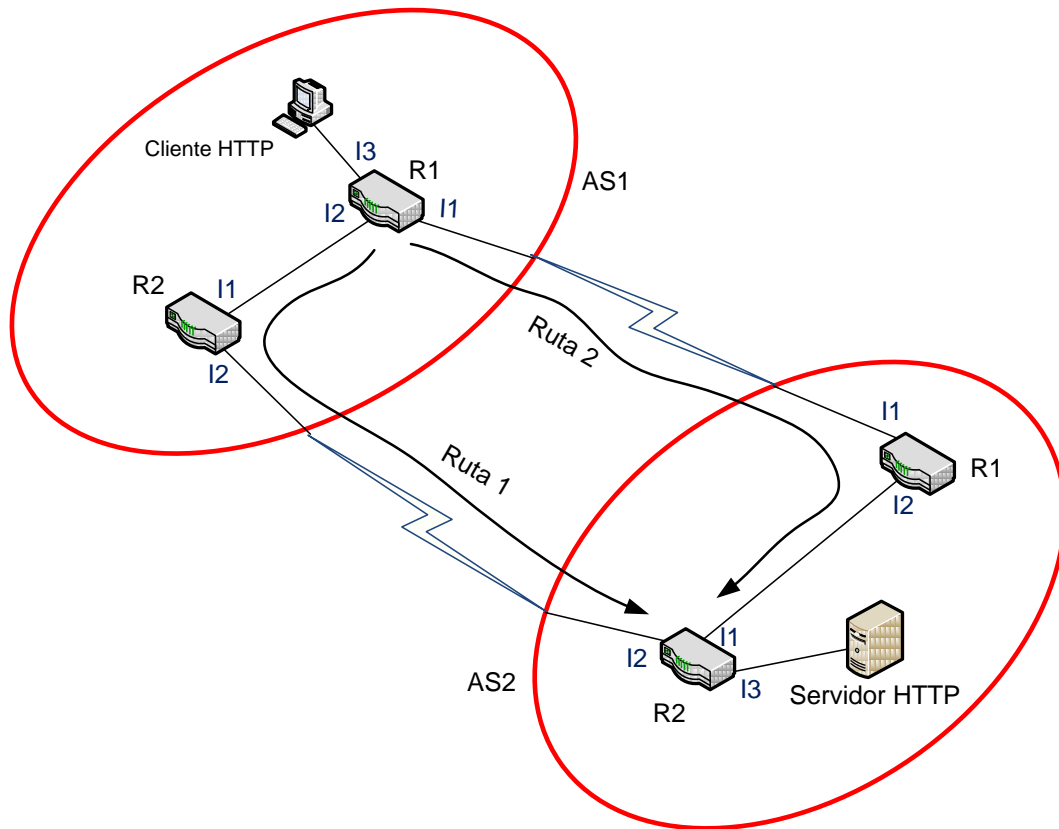


Figura 8 Escenario de simulación 4

```

_schema
[
  _find .schemas.Net
]

Net
[
  frequency 1000000000

  randomstream
  [
    generator "MersenneTwister"
    stream "seedstarter2"
    reproducibility_level "host"
  ]

  traffic
  [

```



```
pattern
  [
    client 1:3
    servers
    [
      list "foobar" nhi 2:3(0) port 80
    ]
  ]
]

bgpoptions
[
  validation_test 12
  startup_jitter_bound 0.1
]

Net
[
  id 1
  AS_status boundary
  router
  [
    id 1
    graph
    [
      ProtocolSession
      [
        name bgp
        use SSF.OS.BGP4.BGPSession
        autoconfig false
        connretry_time 120
        min_as_orig_time 15
        reflector false
        neighbor
        [
          as 2
          address 1(1)
          use_return_address 1(1)
          hold_time 90
          keep_alive_time 30
          mrai 30
          infiltrer
          [
            clause
            [
              precedence 1.
              predicate []
              action
              [
                primary permit
              ]
            ]
          ]
        ]
      ]
    ]
  ]
]
```

```
    ]
  ]
  outfilter
    [
      _extends .filters.permit_all
    ]
  ]
  neighbor
  [
    as 1
    address 2(1)
    use_return_address 1(2)
    hold_time 90
    keep_alive_time 30
    mrai 30.
    infiltrer
    [
      clause
      [
        precedence 1
        predicate []
        action
        [
          primary permit
        ]
      ]
    ]
  ]
  outfilter
  [
    _extends .filters.permit_all
  ]
  ]
  ProtocolSession
  [
    name socket
    use SSF.OS.Socket.socketMaster
  ]
  ProtocolSession
  [
    name tcp
    use SSF.OS.TCP.tcpSessionMaster
    warn false
  ]
  ProtocolSession
  [
    name ip
    use SSF.OS.IP
  ]
  ]
  interface
```

```
[
  id 0
  virtual true
]
interface
[
  idrange
  [
    from 1 to 3
  ]
]
]

router
[
  id 2
  graph
  [
    ProtocolSession
    [
      name bgp
      use SSF.OS.BGP4.BGPSession
      autoconfig false
      connretry_time 120
      min_as_orig_time 15
      reflector false
      neighbor
      [
        as 2
        address 2(2)
        use_return_address 2(2)
        hold_time 90
        keep_alive_time 30
        mrai 30
        infilter
        [
          clause
          [
            precedence 1
            predicate []
            action
            [
              primary permit
            ]
          ]
        ]
      ]
      outfilter
      [
        _extends .filters.permit_all
      ]
    ]
  ]
]
```

```
neighbor
[
  as 1
  address 1(2)
  use_return_address 2(1)
  hold_time 90
  keep_alive_time 30
  mrai 30
  infiltr
  [
    clause
    [
      precedence 1
      predicate []
      action
      [
        primary permit
      ]
    ]
  ]
  outfilter
  [
    _extends .filters.permit_all
  ]
]
]
ProtocolSession
[
  name socket
  use SSF.OS.Socket.socketMaster
]
ProtocolSession
[
  name tcp
  use SSF.OS.TCP.tcpSessionMaster
  warn false
]
ProtocolSession
[
  name ip
  use SSF.OS.IP
]
]
interface
[
  id 0
  virtual true
]
interface
[
  idrange
```

```

    [
      from 1 to 2
    ]
  ]
]

# Cliente HTTP
host
[
id 3
interface
  [
  id 0
  bitrate 100000000
  latency 0.0
  ]
  nhi_route
  [
  dest 2:3(0)
  interface 0
  next_hop 1:1(3)
  ]
  graph
  [
  ProtocolSession
  [
  name client
  use SSF.OS.WWW.httpClient
  start_time 35.0
  inter_session_time
  [
  distribution
  [
  name "Exponential"
  lambda 0.01
  ]
  ]
  ]
  pages_in_session
  [
  distribution
  [
  name "Exponential"
  lambda 0.2
  ]
  ]
  ]
  inter_page_time
  [
  distribution
  [
  name "Pareto"
  k 25.0
  ]
  ]
  ]
  ]
  ]

```

```
        alpha 2.0
      ]
    ]
  inter_request_time
  [
    distribution
    [
      name "Pareto"
      k 0.1667
      alpha 1.5
    ]
  ]
  server_list "foobar"
_find .common.httppinit.http_hdr_size
_find .common.httppinit.show_report
_find .common.httppinit.persistent
_find .common.httppinit.debug
]
ProtocolSession
[
  name socket
  use SSF.OS.Socket.socketMaster
]
ProtocolSession
[
  name tcp
  use SSF.OS.TCP.tcpSessionMaster_find .common.tcpinit
]
ProtocolSession
[
  name ip
  use SSF.OS.IP
]
]
]
]
Net
[
  id 2
  AS_status boundary
  router
  [
    id 1
    graph
    [
      ProtocolSession
      [
        name bgp
        use SSF.OS.BGP4.BGPSession
        autoconfig false
      ]
    ]
  ]
]
```

```
connretry_time 120
min_as_orig_time 15
reflector false
neighbor
  [
  as 1
  address 1(1)
  use_return_address 1(1)
  hold_time 90
  keep_alive_time 30
  mrai 30
  infilter
    [
    clause
      [
      precedence 1
      predicate []
      action
        [
        primary permit
        ]
      ]
    ]
  outfilter
    [
    _extends .filters.permit_all
    ]
  ]
neighbor
  [
  as 2
  address 2(1)
  use_return_address 1(2)
  _extends .basic_ebgp_neighbor
  ]
]
ProtocolSession
[
name socket
use SSF.OS.Socket.socketMaster
]
ProtocolSession
[
name tcp
use SSF.OS.TCP.tcpSessionMaster
warn false
]
ProtocolSession
[
name ip
use SSF.OS.IP
```

```
]
]
interface
  [
    id 0
    virtual true
  ]
interface
  [
    idrange
    [
      from 1 to 2
    ]
  ]
]

router
  [
    id 2
    graph
    [
      ProtocolSession
      [
        name bgp
        use SSF.OS.BGP4.BGPSession
        autoconfig false
        connretry_time 120
        min_as_orig_time 15
        reflector false
        neighbor
        [
          as 1
          address 2(2)
          use_return_address 2(2)
          hold_time 90
          keep_alive_time 30
          mrai 30
          infiltrer
          [
            clause
            [
              precedence 1
              predicate []
              action
              [
                primary permit
              ]
            ]
          ]
        ]
      ]
      outfilter
      [
```



```
        _extends .filters.permit_all
    ]
]
neighbor
[
  as 2
  address 1(2)
  use_return_address 2(1)
  _extends .basic_ebgp_neighbor
]
]
ProtocolSession
[
  name socket
  use SSF.OS.Socket.socketMaster
]
ProtocolSession
[
  name tcp
  use SSF.OS.TCP.tcpSessionMaster
  warn false
]
ProtocolSession
[
  name ip
  use SSF.OS.IP
]
]
interface
[
  id 0
  virtual true
]
interface
[
  idrange
  [
    from 1 to 3
  ]
]
]

# Servidor HTTP
host
[
  id 3
  interface
  [
    id 0
    bitrate 100000000
    latency 0.0
  ]
]
]
```

```
]
nhi_route
[
dest 1:3(0)
interface 0
next_hop 2:2(3)
]
graph
[
ProtocolSession
[
name server
use SSF.OS.WWW.httpServer
port 80
client_limit 10
objects_in_page
[
distribution
[
name "Pareto"
k 0.6667
alpha 1.2
]
]
]
object_size
[
distribution
[
name "Pareto"
k 2000.0
alpha 1.2
]
]
]
response_delay
[
distribution
[
name "Exponential"
lambda 10.0
]
]
]
_find .common.httpinit.http_hdr_size
_find .common.httpinit.show_report
_find .common.httpinit.debug
]
ProtocolSession
[
name socket
use SSF.OS.Socket.socketMaster
]
ProtocolSession
```

```
[
  name tcp
  use SSF.OS.TCP.tcpSessionMaster_find .common.tcpinit
]
ProtocolSession
[
  name ip
  use SSF.OS.IP
]
]
]

link
[
  attach 1:1(1)
  attach 2:1(1)
]
link
[
  attach 2:1(2)
  attach 2:2(1)
]
link
[
  attach 2:2(2)
  attach 1:2(2)
]
link.
[
  attach 1:2(1)
  attach 1:1(2)
]
link
[
  attach 1:1(3)
  attach 1:3(0)
]
link
[
  attach 2:2(3)
  attach 2:3(0)
]
]

common
[

  tcpinit
  [
    ISS 10000
```

```
MSS 2000
RcvWndSize 32
SendWndSize 32
SendBufferSize 128
MaxRexmitTimes 12
TCP_SLOW_INTERVAL 0.5
TCP_FAST_INTERVAL 0.2
MSL 60.0
MaxIdleTime 600.0
delayed_ack false
fast_recovery true
show_report false
debug false
]

httpinit
[
  http_hdr_size 1000
  persistent false
  show_report true
  debug false
]
]
```

BIBLIOGRAFIA

- [1] Material de estudio, “*Practica II: Introducción a OPNET*,” Departamento de Telemática y Computación. Universidad de Oviedo, Madrid, España. [Online]. Disponible en: <http://www.it.uniovi.es/old/material/telematica/ars/ars-prac2.pdf> [Ingreso, Abril 2009]
- [2] “*The Network Simulator – ns2*,” Instituto de Ciencias de la Información. Universidad de California del Sur. Disponible en: <http://www.isi.edu/nsnam/ns/>. [Ingreso, Abril 2009]
- [3] M. Herrera y M. José, “*NS-2 Network Simulator*,” Valparaíso, 12 de Mayo de 2004.
- [4] “*How To Use SSFNet*,” The SSF Research Network. Disponible en: <http://www.ssfnet.org/internetPage.html> [Ingreso, Noviembre 2008]
- [5] B. Premore, “*SSFNet And Routing Simulation*,” Instituto de Matemáticas Puras y Aplicadas, Universidad de California, Los Ángeles, California, EU. [Online]. Disponible en: http://www.ipam.ucla.edu/publications/cntut/cntut_1501.ppt. [Ingreso, Noviembre 2008]
- [6] Y. Rekhter, T. Li, y S. Hares. “*A Border Gateway Protocol 4 (BGP-4)*,” Internet Society RFC4271. IETF. Enero 2006. [Online]. Disponible en: <http://www.ietf.org/rfc/rfc4271.txt> [Ingreso, Abril. 2009].
- [7] Material de estudio, “*El JDK (Java Development Kit)*,” Área de Lenguajes y Sistemas Informáticos. Universidad de Burgos, España. [Online]. Disponible en: http://pisuerga.inf.ubu.es/lsi/Invest/Java/Tuto/A_.htm [Ingreso, Septiembre 2009].
- [8] “*Code Samples and Apps, Applets*,” SUN Microsystems, Inc. [Online]. Disponible en: <http://java.sun.com/applets/> [Ingreso, Septiembre 2009].
- [9] J. Castaño, J. Rojas. “Estudio de Viabilidad para la Optimización de Enrutamiento IP con el Protocolo BGP”. Tesis. Universidad Del Cauca. Popayán, Colombia. Noviembre, 2008.
- [10] “Como Citar Referencias”. *IEEE Style*, Julio, 2008. [Online]. Disponible en: <ftp://ftp.unicauca.edu.co/Facultades/FIET/IPET/IEEE-Style.pdf> [Ingreso, Noviembre. 2008].