UACM
Universidad Autónoma de la Ciudad de México
Nada humano me es ajeno

COLEGIO DE CIENCIA Y TECNOLOGÍA

LICENCIATURA EN INGENIERÍA EN SISTEMAS ELECTRÓNICOS Y DE TELECOMUNICACIONES

Análisis de gestión en la red avanzada europea GEANT

T E S I S

PARA OPTAR POR EL TITULO DE

LICENCIADO EN INGENIERÍA EN SISTEMAS ELECTRÓNICOS Y DE TELECOMUNICACIONES

P R E S E N T A

FERNANDO DE LA CRUZ ALEJANDRE

D I R E C T O R

M. EN C. JOSÉ IGNACIO CASTILLO VELÁZQUEZ

Ciudad de México, junio de 2018

UACM
Universidad Autónoma de la Ciudad de México
Nada humano me es ajeno

CA Coordinación Académica

SISTEMA BIBLIOTECARIO DE INFORMACIÓN Y DOCUMENTACIÓN

UNIVERSIDAD AUTÓNOMA DE LA CIUDAD DE MÉXICO
COORDINACIÓN ACADÉMICA

RESTRICCIONES DE USO PARA LAS TESIS DIGITALES

DERECHOS RESERVADOS©

Agradecimientos

Gracias a dios por permitirme esta oportunidad fantástica de vida, rodeado de experiencias maravillosas y de mis seres queridos. Agradezco eternamente a mis padres Olga Alejandre Echeverría y Fernando Ladislao de la Cruz Martínez, por su paciencia, motivación, apoyo incondicional y su precioso cariño que me permitieron llegar hasta aquí y comenzar una nueva etapa de vida.

Agradezco a mis hermanos Víctor Alfonso, María del Rosario y Mario Alberto, que a pesar de adversidades nos mantenemos unidos, a mis tíos y abuelos, por sus experiencias transmitidas, a mis padrinos, por el apoyo moral, a mis primos, sobre todo a mi primo Federico Alejandre Becerra por su apoyo y motivación incondicional.

Agradezco a mis seres queridos, que ya no están conmigo, en especial a mi tía Amelia Alejandre Echeverría por todo su apoyo como tía y madre, a mi tío José Isabel Alejandre Echeverría, por darme la tranquilidad de tener un hogar y por su cariño excepcional, por sus enseñanzas que me permitieron ser mejor persona y mantener el orden y disciplina en mi vida. Les agradezco infinitamente a ustedes dos, porque sin ustedes quizá esto no hubiera sido posible. Su historia de humildad, hermandad y lucha y sobre todo ese instante de decisión, que el destino les puso en sus manos, porque con solo $20 pesos cambiaron el destino de muchas personas.

Agradezco a Ana Karen Aguirre Hernández, por compartir una parte de su vida conmigo y apoyarme en todo momento, por todos esos momentos de felicidad, motivación y cariño, Ot Ogtma. A su madre María del Roció Hernández Zavala, por todo el apoyo que me ha brindado para llegar hasta este momento.

Agradezco a mi profesor y director de tesis José Ignacio Castillo Velázquez por sus valores y enseñanzas, por ser la persona que me ayudo a proyectar mi profesión y agregar valor a mi vida.

Agradezco a mis lectores, el Doctor Gerardo Abel Laguna Sánchez, el Maestro Joel Yazbek Buendía Gómez y el Licenciado Ricardo Galindo Reyes, por brindarme el tiempo necesario y compartirme sus puntos de vista.

Agradezco a mi Universidad Autónoma de la Ciudad de México, por darme la oportunidad de encontrar a esas personas gratas en mi vida, amig@s, profesores que me dejaron experiencias, pero, sobre todo por brindarme la oportunidad de titularme como ingeniero en Sistemas Electrónicos y de Telecomunicaciones.

Agradezco a servicios estudiantiles de la Universidad Autónoma de la ciudad de Mexico, por apoyarme económicamente para la impresión de este tomo.

Contenido

Resumen

GEANT es una de las redes avanzadas más importantes del mundo, dedicada a la comunicación entre comunidades académicas y de investigación europeas. Su evolución desde el año 2000 hasta la actualidad, ha permitido ofrecer una red de alta velocidad, con ancho de banda de hasta 500 Gbps, lo que ha permitido el desarrollo de importantes proyectos como el Gran Colisionador de Hadrones, encargado del estudio de la física de partículas, el proyecto ILLUSTRIS, dedicado a la simulación más completa del universo, entre otros. Estos, exigen la necesidad de compartir y distribuir información en grandes volúmenes a distintas partes de Europa y el resto del mundo. Su red provee de servicio aproximadamente 50 millones de usuarios entre 41 países europeos, con un tráfico de 4000 TB por día.

El objetivo de este trabajo es analizar la topología de GEANT de enero de 2017 a nivel de red, para probar conectividad y gestión en la capa de transmisión del Backbone, mediante la simulación, emulación de red y aplicando los protocolos OSPF y SNMP.

En la simulación de la red GEANT, se logró configurar OSPF v2, comprobando conectividad con éxito. También, se realizó gestión de la red por medio de herramientas basadas en el protocolo SNMP v2. Sin embargo, no fue posible configurar mensajes de alerta llamados *Traps* ni SNMP v3, la cual ofrece autentificación y cifrado en los mensajes de gestión. Por otro lado, en la emulación de GEANT se logró configurar OSPV v2, SNMP v2 y SNMP v3, lo que permitió configurar la autentificación MD5 y cifrado DES para la seguridad de los mensajes de gestión y tanto la conectividad como la gestión fueron comprobadas satisfactoriamente.

La simulación y emulación del Backbone GEANT, se realizó en un equipo portátil Sony VAIO, con procesador Intel Core(TM) i5-2410M, CPU 2.30 GHz, RAM de 12 GB y disco de estado sólido de 120 GB. En la simulación se registró un uso de recursos de 10% de CPU y 2.025% de RAM, mientras que en la emulación se registró un uso de 100% de CPU y 55% de RAM, en otras palabras, la emulación requirió más de 9 veces el uso de CPU y más de 27 veces el uso de RAM en comparación con la simulación.

Capítulo 1

Introducción

Hoy en día, las telecomunicaciones se han convertido en una necesidad de dominio público ya que estas son una herramienta fundamental para la comunicación y el desarrollo de comunidades y países enteros. Nuestro contexto actual y el de hace unos años nos permiten hacer un contraste de las entre las comunicaciones que se realizaban hace 30 años y las que hoy en día se realizan. Un claro ejemplo es la red de Internet, una herramienta que nos permite prescindir de tecnologías que antes eran necesarias para la comunicación. Esta red nos permite acceder, compartir y almacenar información, además de poder ofrecer comunicación entre personas en tiempo real desde cualquier parte del mundo, sin embargo, la red de Internet ha sido un proceso de desarrollo y evolución de las redes de datos.

Desde 1964 cuando ARPANET (*Advanced Research Projects Agency Networks*) solidificó los trabajos de Leonard Kleinrock, Paul Baran y Donald Watts Davis para proponer una red con base en la conmutación de paquetes, la cual, cambió el paradigma de la conmutación de circuitos y dio inició a un proyecto por parte de DARPA (*Defense Advanced Research Projects Agency*) para conectar distintos campus de universidades en los EEUU. En 1969 la UCLA (*University of California Los Ángeles*), creó la primera red LAN (*Local Área Network*) más grande del mundo, con 30 estaciones informáticas conectadas con un modem a 56 Kbps vía PSTN (*Public Switched Telephone Network*). No obstante, en ese mismo año se desarrollaron los conmutadores de paquetes llamados lMP (*Interfaz Message Processor*), que permitieron conectar las redes LAN de las universidades de UCLA, SRI (*Standford Research Institute*) con la UCSB (*University of California- Santa Bárbara,)*, para dar origen a la primera red MAN (*Metropolitan Area Network*) más grande del mundo y a finales de 1969 se conectó la universidad de UTAH con la red MAN, dando origen a la primera red de datos basada en conmutación de paquetes y conectada mediante IMP (*Interface Message Processor*), que eran computadoras programadas mediante NCP (*Network Control Program*) para la conexión entre computadoras de las distintas universidades [1, 2].

Por su parte en la década de los setenta, se marcó el inicio y desarrollo de las primeras redes europeas. Tiempo después, se dio la estandarización de las redes en todo el mundo con el protocolo TCP/IP y a principios de los noventa surgieron las primeras NREN (*National Research and Education Network*), dedicadas a la comunicación entre centros de académicos y de investigación entre países. Posteriormente en Europa, hubo proyectos como EUROPANET y TEN-35 que promovieron el desarrollo de las redes avanzadas europeas, que se caracterizaron por proveer de infraestructura y tecnología para ofrecer redes de alta velocidad. Desde entonces estos proyectos han evolucionado hasta a la red GEANT, la cual actualmente proporciona conectividad en Gbps a las NREN europeas.

En este trabajo se estudiará el funcionamiento de la red GEANT, mediante procesos de simulación y emulación, para lo cual es indispensable abordar los protocolos de enrutamiento, conocer sus características, ventajas y desventajas, posteriormente analizaremos el protocolo de gestión SNMP para tener los fundamentos teóricos y lograr aplicar la gestión en la red en simulación y emulación con autentificación y cifrado.

Las herramientas necesarias para llevar a cabo este trabajo, serán procesos de simulación y emulación de red, mediante programas como Packet Tracer y GNS3. Estos nos brindarán un ambiente didáctico y grafico para aproximar el funcionamiento físico y lógico del Backbone de GEANT, realizando pruebas de conectividad, además de implementar la gestión en los elementos de red, con base en el protocolo SNMP v2 y v3 usando autentificación y cifrado. Por último, se explicarán los resultados obtenidos en cada uno de los procesos realizados.

Justificación

Los avances tecnológicos de Internet han permitido que día a día se puedan realizar transferencias de datos de manera eficiente, confiable y segura, sin embargo, estos cambios aún no son suficientes para proveer de conectividad a las comunidades académicas y de investigación, ya que estas generan y comparten información en grandes volúmenes día a día, exigiendo una red de alta velocidad, que satisfaga dichas necesidades.

Dado el contexto actual de la red de Internet y su importancia para la comunicación entre personas, empresas y gobiernos en todo el mundo y a su vez las redes avanzadas para el desarrollo de proyectos entre investigadores y académicos, es necesario tener conocimientos solidos del funcionamiento de las redes de datos, que nos permitan implementar herramientas necesarias de administración y gestión de red, por medio de protocolos y estándares, para ofrecer cada vez mejores redes con alta disponibilidad, fiabilidad y seguridad.

Objetivo General

Estudiar mediante simulación y emulación la red europea GEANT, para contribuir y complementar los estudios ya realizados sobre las redes CUDI (Corporación Universitaria para el Desarrollo de Internet), CLARA (Cooperación Latino Americana de Redes Avanzadas), CANARIE (*Canada Advanced Research and Innovation Network*) e INTERNET 2, en el ADVNETLAB (*Advanced Networking Laboratory*) de la Universidad Autónoma de la Ciudad de México.

Objetivos específicos

1. Conocer la historia y evolución de las redes en Europa y sus redes avanzadas.
2. Conocer el funcionamiento de la red de Internet y de una Red Avanzada.
3. Estudiar y analizar a nivel de red, la conectividad y gestión en la simulación y emulación del Backbone de GEANT.
4. Poner a prueba los simuladores y emuladores de red (Packet Tracer y GNS3) para crear un ambiente de tipo Backbone lo más cercano a la realidad usando la topología de GEANT implementada en 2017.
5. Implementar los protocolos de enrutamiento RIP v2, OSPF v2 y de gestión SNMP (v2 y v3) en la red GEANT.
6. Conocer el funcionamiento y operación de los protocolos RIP v2 y OSPF en una red Gigabit.
7. Gestionar la red GEANT por medio del protocolo SNMP v2 y SNMP v3.
8. Conocer las ventajas y desventajas de los procesos de simulación y emulación de redes.
9. Desarrollar las habilidades técnicas y no técnicas de un administrador de red.

Distribución de capítulos

Este trabajo se compone de 6 capítulos: en el capítulo 1 “Introducción”, se explica el inicio de las primeras redes de datos.

El capítulo 2 “Redes de datos en Europa y sus redes avanzadas”, explica el inicio de las primeras redes de datos europeas con el proceso de estandarización del protocolo TCP/IP, las conexiones transatlánticas entre EEUU y Europa, el origen de las NREN europeas y la evolución de sus redes avanzadas hasta llegar a la red GEANT.

El capítulo 3 “Protocolos de Enrutamiento y de Gestión de red”, aborda a detalle los protocolos de enrutamiento RIP y OSPF, fundamentales para la conectividad de red y el protocolo de gestión SNMP para la gestión de red.

En el capítulo 4 “Simulación y Emulación de la red GEANT”, se implementa la topología del Backbone de GEANT en simulación y emulación, configurando los protocolos RIP v2, OSPF v2 y SNMP v2 y v3, para probar gestión aplicando autentificación y cifrado.

El capítulo 5 “Resultados”, describe todas las pruebas realizadas en la simulación y emulación desde pruebas de conectividad, análisis de paquetes ARP, ICMP, OSPF y SNMP, pruebas de conectividad

simultaneas, pruebas de métricas, de latencia y convergencia entre RIP v2 y OSPF v2, pruebas de fallo de router con tráfico de paquetes, monitoreo de la red con SNMP v2 y v3, gestión de la red con SNMP v2 y v3 e implementación de cifrado y autentificación de SNMP v3.

Finalmente, en el capítulo 6 “Conclusiones” se hace una comparativa entre los objetivos y los resultados alcanzados.

Capítulo 2

Redes de datos en Europa y sus redes avanzadas

2.1. Conexión de ARPANET con Europa

Dado el éxito de la red ARPANET en EEUU, en el año 1973 se creó un proyecto para realizar la primera conexión de red con Europa. Esta se realizó entre el SDAC en Virginia y el NORSAR (*Norwegian Seismic Array*) de Noruega, vía satelital y posteriormente se conectó a Londres por medio del uso de IMP, para crear una red compartida de información de actividad sísmica y nuclear entre EEUU y Europa como se muestra en la figura 1.

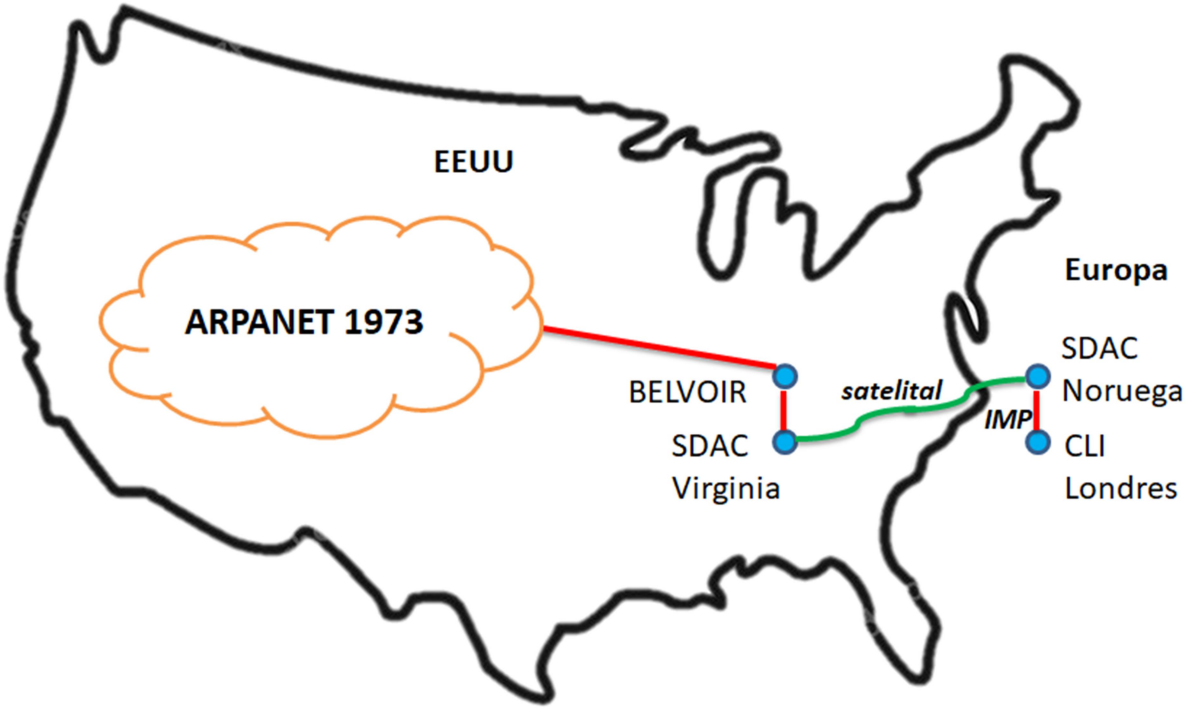


Figura 1. Primera conexión de red entre EEUU y Europa.

En contraste con las redes de EEUU, las redes en Europa en su mayoría eran centros informáticos de bajo impacto. No obstante, hubo redes europeas que tuvieron avances y aportaciones considerables como la red CERNET de UK, NORDUNET (*Nordic Infraestructure for Research and Education)*, el CERN (*Conseil Européen pour la Recherche Nucléaire*) y la red EARN (*European Academic and Research Network*), impulsada por la empresa IBM para el desarrollo y distribución de las redes en Europa [1, 3, 4].

2.1.1. Protocolo TCP/IP

El objetivo de conectar EEUU con el continente europeo, fue estandarizar las redes de datos de ambos continentes por medio de un protocolo de conmutación de paquetes. En el año de 1974 Vinton Cerf y Robert Kahn publicaron una propuesta con base en un conjunto de protocolos de Ethernet para la intercomunicación de paquetes de red y después de realizar pruebas satelitales y de radio, surgió el protocolo TCP (*Transmission Control Protocol*). En contraste en 1976, la organización de normalización CCITT (*Consultative Committee for International Telegraphy and Telephony*) inició el desarrollo de un protocolo para redes conocido como X.25, mismo que adoptaron las redes europeas, sin embargo, dada la diversidad de protocolos de conmutación en el año 1979, ISO (*International Organization for Standardization*) publicó el modelo de referencia OSI (*Open **S**ystem **I**nterconnection*) como un modelo para protocolos red que pudiera garantizar la compatibilidad entre las computadoras de red. A pesar de ello en 1980, se liberó por completo el protocolo TCP y tres años después el protocolo TCP/IP (*Transmision Control Protocol- Internet Protocol*) se declaró como un estándar para las redes de datos [1, 3-6].

2.1.2. Protocolo X.25 en Europa

A pesar de las ventajas que ofreció el protocolo TCP/IP, las redes europeas se negaron a usarlo, dado que pensaron que era un protocolo incompleto y transitorio para las redes de datos, por lo que adoptaron el protocolo X.25, con la idea de tener sus propios protocolos de comunicación y competir con las redes de EEUU. Este contexto permitió que algunas redes europeas propusieran protocolos de comunicación, tal fue el caso de la red CERNET de UK, la cual propuso estándares y protocolos de red con el nombre de "*Colors Book*", desafortunadamente no lograron alcanzar los estándares de la norma ISO y sólo pudieron aplicarse dentro de su red. También, con un papel fundamental en la historia de las redes de Europa, el centro de investigación nuclear más importante del mundo el CERN, tuvo la necesidad de crear sus propios estándares y protocolos de red, por medio de sistemas de conmutación llamados INDEX con terminales de Gandalf Tec. [4, 5]

2.1.3. IBM en las primeras redes de Europa

La introducción del protocolo TCP/IP, permitió que IBM realizara grandes contribuciones en el desarrollo y evolución de las redes europeas. Un ejemplo claro fue el financiamiento de la red EARN, en el año de 1983 con la aportación de tecnologías y protocolos IBM (RSCS/NJE), logró conectar a poco más de mil computadoras en su mayoría *mainframes* IBM con sistemas operativos VM/CMS (*Virtual Machine* /*Conversacional Monitor System*) ubicados en centros de universidades e

instituciones de investigación distribuidos por Roma, Italia y Bonn ciudad de Alemania, además de lograr una conexión transatlántica entre la red EARN en Europa y la red BINET de EEUU por medio del protocolo TCP/IP. [1, 4,5]

2.1.4. La estandarización del protocolo TCP/IP en Europa y el inicio de la red NORDUNET

A pesar de que el protocolo X.25 fue un estándar para las redes en Europa, a mediados de la década de los ochenta el protocolo TCP/IP empezó a introducirse por el norte de Europa. Las redes académicas de Suecia, Noruega, Finlandia, Dinamarca e Islandia, formaron redes troncales hibridas, con una variedad de tecnologías y protocolos de comunicación que proporcionaron conectividad a poco más de cien mil computadoras para dar paso a la red llamada NORDUNET. Su infraestructura soportó protocolos como, X.25, DECNET, TCP/IP, ISO CONS, y utilizó tecnologías Ethernet y Cisco "*Multiprotocol Router"* como se muestra en la figura 2. [4,5]

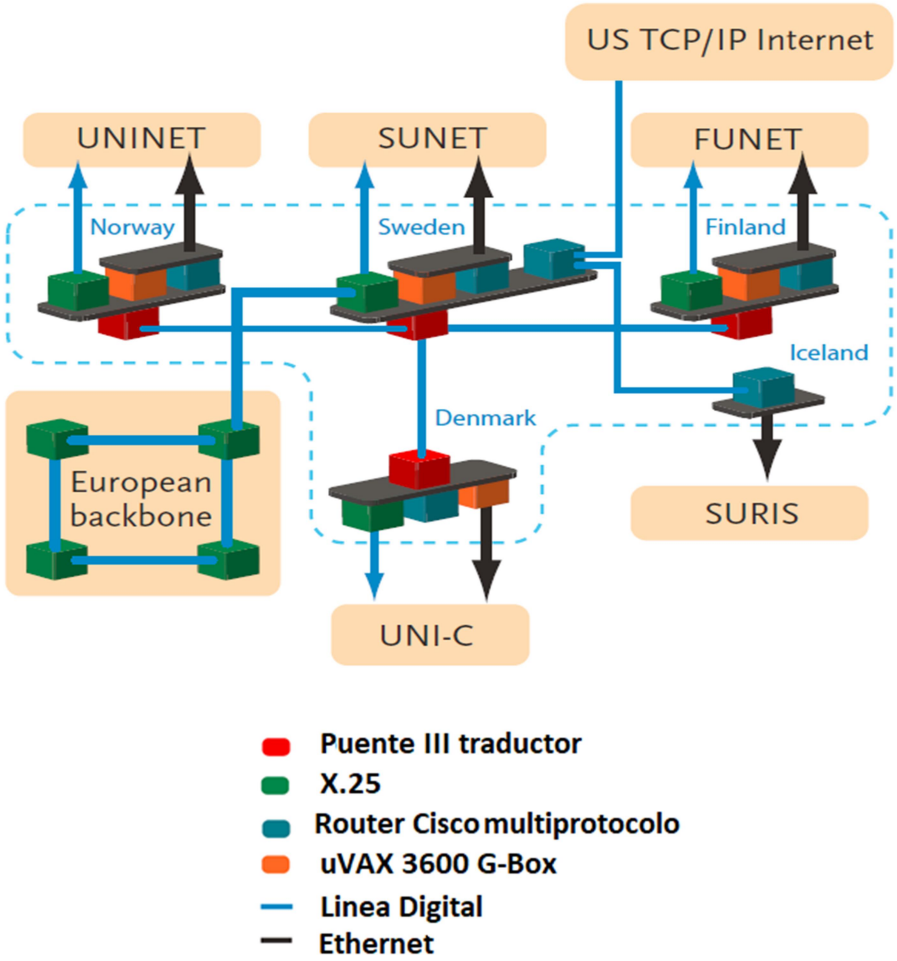


Figura 2.Topología de las redes Nórdicas [4].

2.2. Inicio de las NREN europeas

A finales de la década de los ochenta, el desarrollo y la expansión del protocolo TCP/IP en el continente europeo permitió crear acuerdos como RARE (*Reseaux Associes pour la Recherche Europeenee*), con el objetivo de acelerar la estandarización de las redes entre ambos continentes. Por otro lado, en paralelo nació COSINE (*Cooperation for OSI Networking in Europe*), que se creó dentro del Programa "Eureka" de la Comunidad Europea en el periodo de 1987-1993, su objetivo fue crear una Infraestructura de red IP para toda la comunidad académica y de investigación europea. Los resultados de este proyecto fueron conexiones de 64Kbps entre las redes europeas, conectividad internacional a Rumania, República Checa y Hungría, además de proporcionar hardware y software a redes de 11 países de Europa central y oriental. Por último, en noviembre de 1989 nació RIPE (*Résseaux IP Européens*) un proyecto para coordinar diferentes aspectos técnicos y administrativos necesarios para garantizar la correcta operación y expansión de la red IP en Europa.

Los proyectos anteriores rindieron frutos a finales de la década de los ochenta al conectar las redes europeas NORDUNET, INRIA (*Institut National de Recherche en Informatique et en Automatique*) de Francia, el Colegio de Londres, El CERN, la Agencia Espacial Europea y el Grupo Europeo de Usuarios de Unix, con instituciones de los EEUU como la NSF (*National Science Foundation*), la NASA (*National Aeronautics and Space Administration*) y el Departamento de Energía (DoE). En el año 989 se logró conexión intercontinental hacia EEUU por Estocolmo, Suecia como se muestra en la figura 3 [1,4, 5-9].

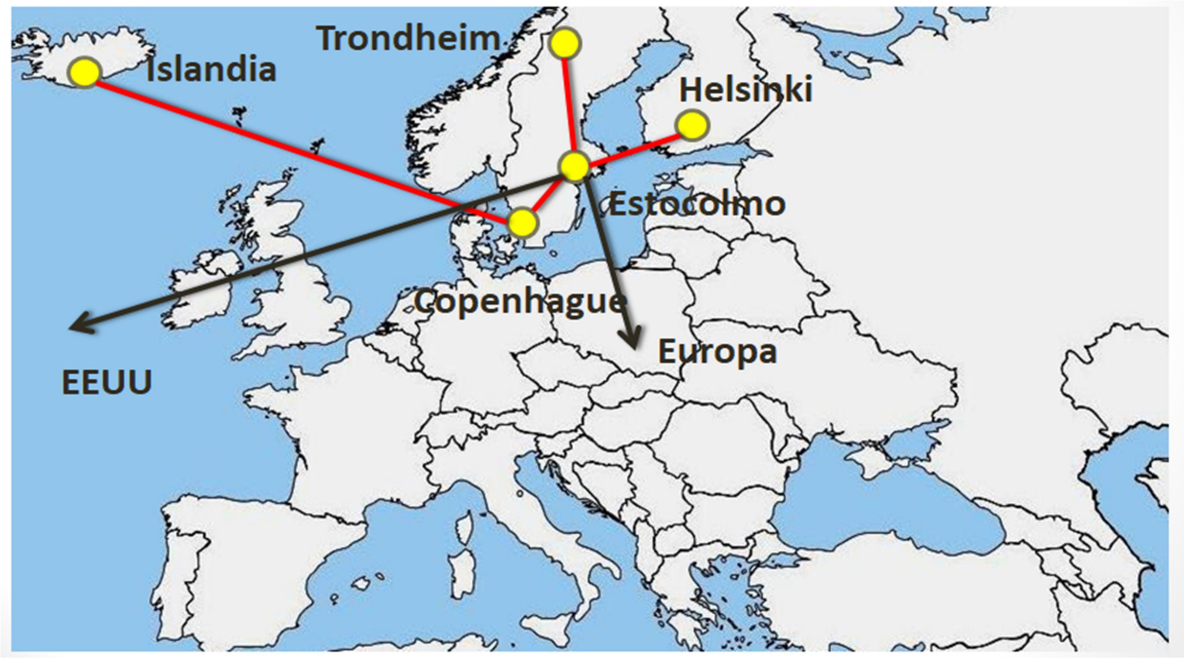


Figura 3. Mapa de interconexión de NORDUNET año 1989.

A principios de la década de los noventa, el ambiente de proyectos para la estandarización de redes europeas, favoreció el inicio formal de las redes de investigación y educación europeas llamadas: NREN y con la fusión de la red EARN y RARE surgió la asociación TERENA (*Trans-European Research and Education Networking Association)*, dedicada a promover, gestionar, redes académicas y de investigación en el continente europeo [5].

2.2.1. Conectividad entre las NREN europeas

En el año de 1991 surgió el primer Backbone Europeo; EBONE (*Europe Backbone*). Su propósito fue establecer una red internacional de interconexión mixta para todas las instituciones académicas, de investigación, redes comerciales y privadas. Brindó conectividad transatlántica entre Amsterdam, Alemania y Washington, EEUU y para el año de 1992 dio paso a una red troncal multiprotocolo denominada EMPB (*European Multi-Protocol Backbone*) usando protocolos X.25 y TCP/ IP, lo que posibilitó tener enlaces IP geográficos más extensos. [10, 11,12]

2.2.2. EUROPANET Y DANTE

EUROPANET fue un proyecto que se creó en el año de 1992 y se puso en marcha en 1993 con un periodo de 4 años, para desarrollar redes europeas que proporcionarían mayor ancho de banda. Este proyecto fue financiado por la Unión Europea y algunos miembros de las NREN europeas como se muestran en la tabla 1.

NREN	PAIS
ACONET	Austria
ARIADNE	Grecia
DFN	Alemania
HEANET	Irlanda
GARR	Italia
NORDUNET	Países Nórdicos
REUNIR	Francia
SURFNET	Suecia
SWITCH	Suiza
VUB-ULB	Bélgica

Tabla 1. Miembros de EUROPANET

EUROPANET en el año de 1994 proporcionó conectividad de hasta 2 Mbps entre las NREN europeas de NORDUNET, DFN de Alemania, JANET de UK, SWITCH de Suiza, REDIRIS de España, las redes de Portugal y Luxemburgo entre otras. También, realizó conexiones transatlánticas hacia EEUU, desde Holanda y el CERN en Suiza como se muestra en la figura 4 [12].

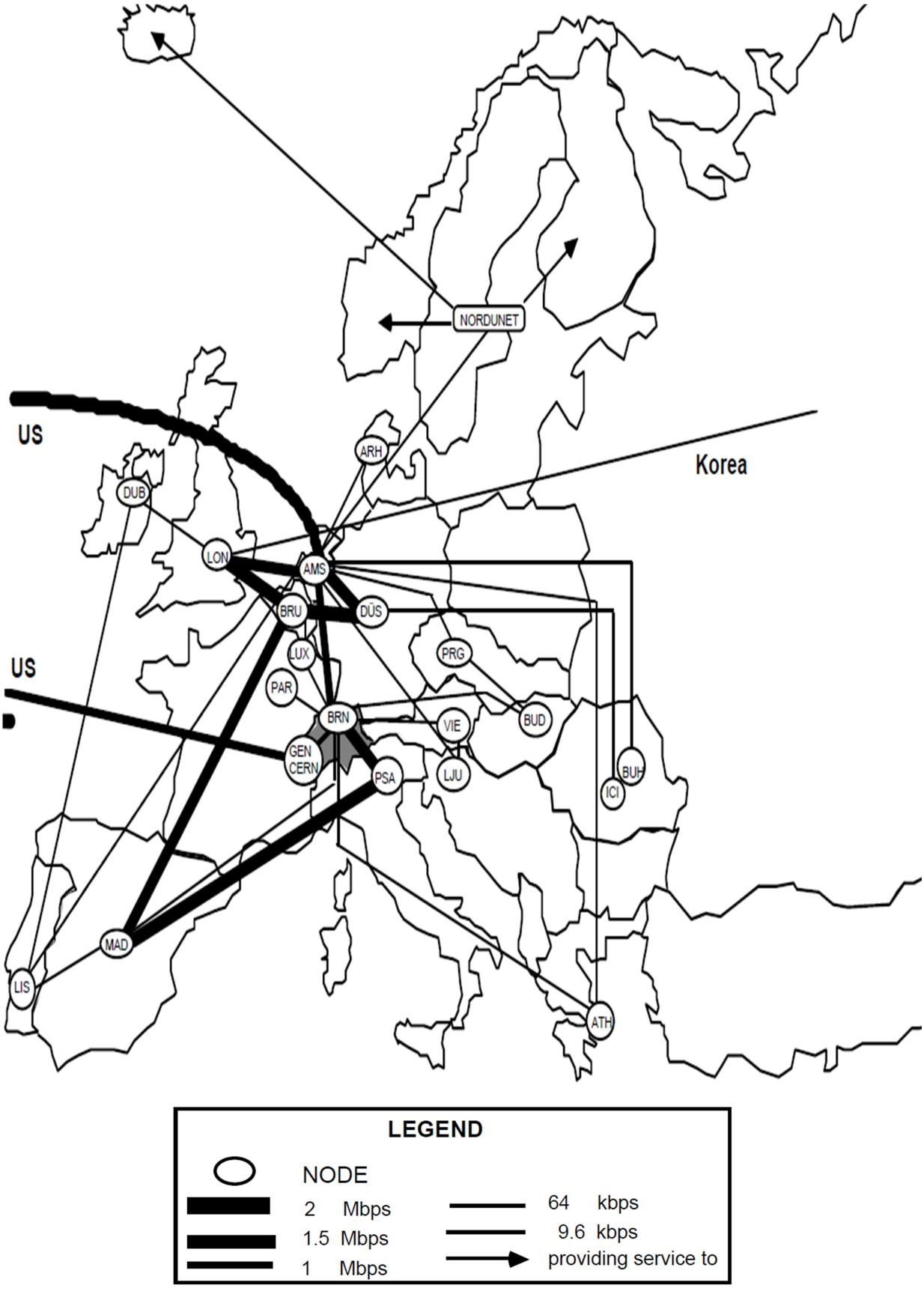


Figura 4. Topología de EUROPANET 1994 [12].

2.2.3. DANTE

DANTE (*Delivery of Advanced Network Technology to Europe)* se creó en 1993 por parte de las NREN europeas y la contribución del proyecto COSINE, con el propósito de proveer y gestionar los servicios de red paneuropeos. Gestionó al Backbone EUROPANET y hoy en día es una sociedad sin fines de lucro que planifica, construye y opera redes de investigación y educación en Europa como la red GEANT [13,14, 15].

2.2.4. EUROCAIRN

En el año de 1993 con la organización de 18 países europeos se creó un proyecto llamado EUROCAIRN (*European Cooperation for Academic and Industrial Research Networking*), el cual junto con DANTE formaron un plan estratégico para la creación de una red de alta velocidad, de 34 Mbps para las NREN de Europa. Los miembros de este proyecto fueron [16, 17, 18]:

- *European Commission*
- *Norway Royal Ministry of Education, Research & Church Affairs (Project Initiator)*
- *Austria Federal Ministry of Science and Research*
- *Belgium Science Policy Office*
- *Denmark Ministry of Research*
- *Finland Ministry of Education*
- *Germany Federal Ministry for Research and Technology*
- *Hungary R&D Information Infrastructure Program Co-ordination Office*
- *Ireland Irish Science and Technology Agency*
- *Italy Ministry of Universities, Scientific Research and Technology (INFN-CNAF)*
- *Luxembourg Ministry of Education*
- *Netherlands Ministry of Education and Science*
- *Portugal Junta Nacional de Investigacao Cientifica e Technologica (JNICT)*
- *Slovenia Ministry of Science and Technology*
- *Sweden Ministry of Education and Science*
- *Switzerland Federal Office of Education and Science*
- *Turkey Scientific and Technical Research Council of Turkey*
- *United Kingdom Joint Information Systems Committe*

2.2.5. El inicio de la red Internet

Una vez que el protocolo IP se expandió a través de las redes europeas, EEUU y el resto del mundo, surgió la necesidad de satisfacer el intercambio de información entre las redes de Investigación y educación no solo en Europa sino de todo el mundo, por lo cual a finales de la década de los ochenta el británico *Tim Berners Lee,* trabajó con idea de fusionar las tecnologías de computadoras por medio de un sistema de información global poderoso y fácil de usar. Sin embargo, fue hasta 1990 cuando *Tim Berners Lee* y el ingeniero *Robert Cailliau* de Bélgica, publicaron una propuesta llamada W.W.W (*World Wide Web*), con la cual podían acceder a documentos en hipertexto de una computadora a otra. Finalmente en el año 1993 Tim Berners Lee y el CERN liberaron la licencia de W.W.W para su uso en todas las redes del mundo, creando una red global, distribuida, descentralizada y unificada por el protocolo de internet y la W.W.W [19, 20].

El 24 de octubre de 1995, el Consejo Federal de Redes FNC (*Federal Networking Council*) de EEUU aprobó por unanimidad una resolución que define el término Internet como el sistema de información global que está lógicamente vinculado entre sí por un conjunto de direcciones globales basadas en el Protocolo TCP/IP. [21]

2.3. Redes Avanzadas europeas

2.3.1. Inicio de las redes de alta Velocidad en Europa: TEN-34 (*Trans-European Network - 34 Mbps*)

Después de la liberación de la red internet en el año 1995, DANTE y la Comisión Europea dentro del proyecto EUROCAIRN, firmaron el acuerdo para crear una nueva red paneuropea llamada TEN-34, misma que, para el año de 1997 conectó a 15 países con velocidades de 2 Mbps hasta 34 Mbps como se muestra en la figura 5 [22].

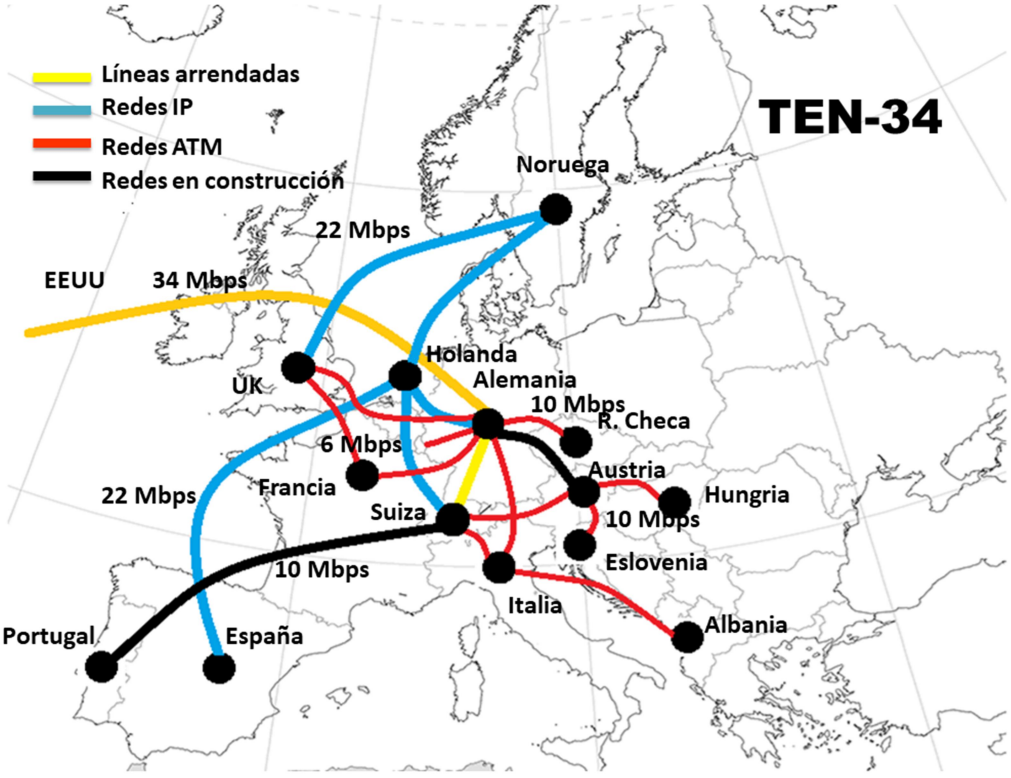


Figura 5. Topología de TEN-34 agosto 1997.

2.3.2. Proyecto QUANTUM

QUANTUM (*Quality Network Technology for User Oriented Multi-Media*), surgió en el año 1997 como una propuesta posterior al proyecto TEN-34. Fue cofinanciado en el marco de la Dirección

General XIII de la Comisión Europea. Considerado como un proyecto transitorio para la red TEN-155 determinó y adquirió la infraestructura de red para sustituir al proyecto TEN-34. También, trabajó con el Proyecto Q-MED del mediterráneo para implementar nuevos protocolos y servicios multimedia en tiempo real para la región Euro-mediterránea [25-27].

2.3.3. TEN-155 (*Trans-European Network- 155 Mbps*)

En diciembre de 1998 surgió la red Trans-Europea TEN-155, la cual representó un gran avance en el ámbito de redes, ya que por primera vez las redes europeas se beneficiaron por la liberalización de las telecomunicaciones, rompiendo la barrera monopólica de precios y el racionamiento de ancho de banda. Su infraestructura de red ofreció conectividad de 35Mbps y hasta 155 Mbps como se muestra en la figura 6 [28, 29].

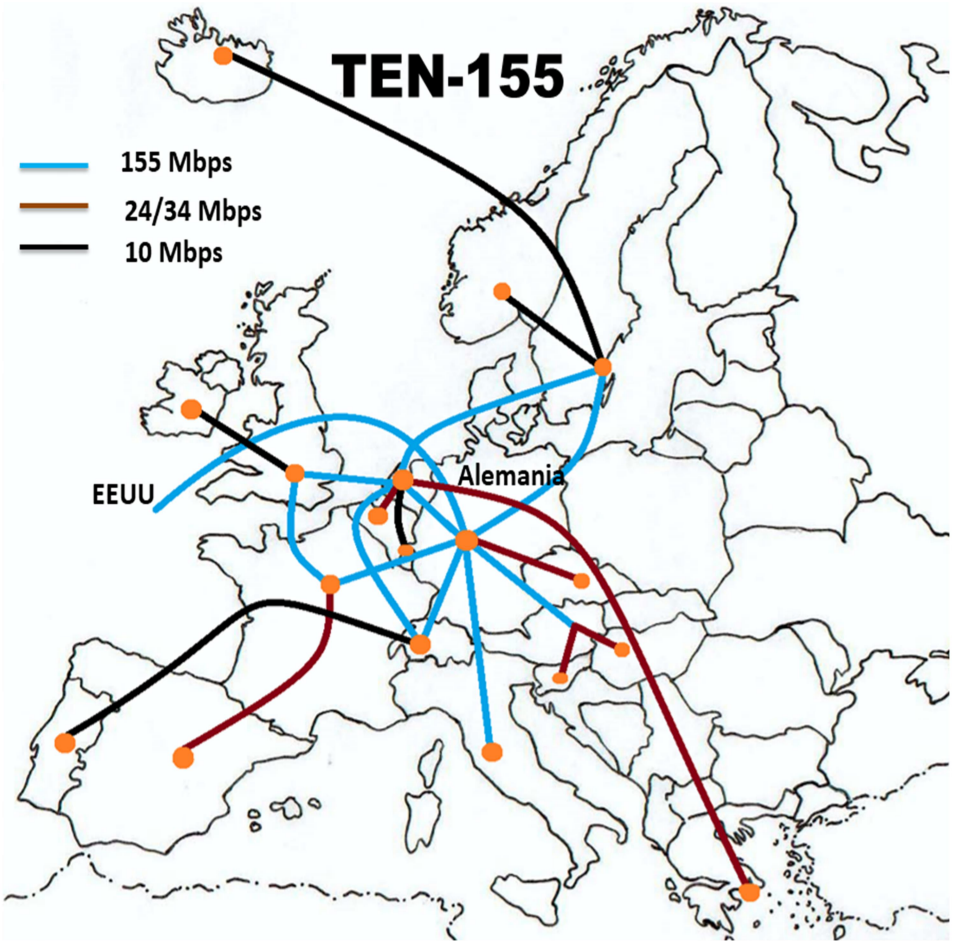


Figura 6. Topología de TEN-155 año 1998.

Para el periodo 1998-2001, TEN-155 alcanzó velocidades de hasta 622Mbps y en mayo de 2001 se conectó con la red SINET (*Science Information Network*) de Japón e Israel como se muestra en la figura 7 [31].

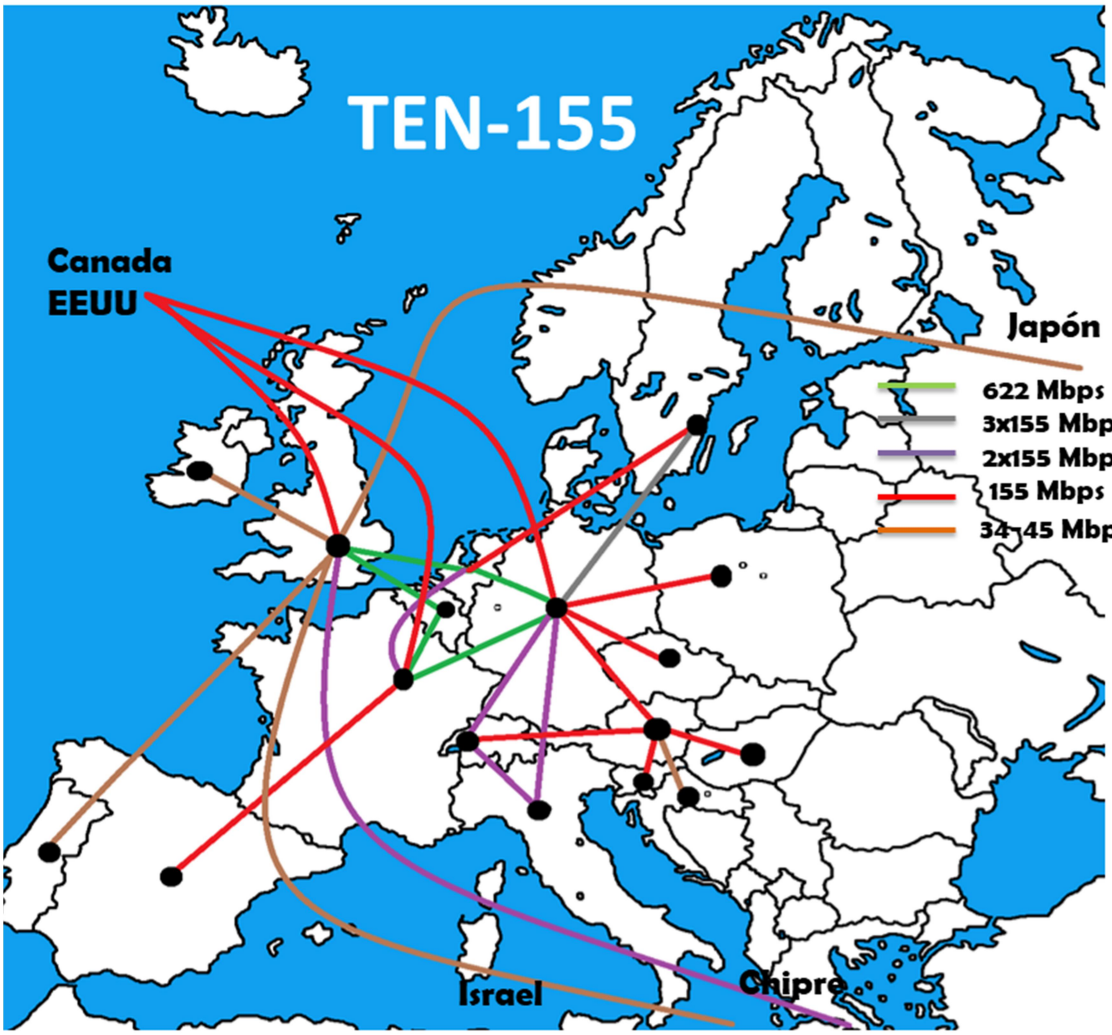


Figura 7. Topología de TEN-155 Mayo de 2001.

2.3.4. Evolución de las primeras redes Europeas NREN y el inicio de las redes Avanzadas

A continuación mostramos la evolución del ancho de banda en las redes europeas y las que ofrecieron los primeros proyectos de redes de alta velocidad. Ver tablas 2-2.1

NREN	AÑO	ANCHO DE BANDA
EARN	1982	9.6 Kbps
SUNET y NORDUNET	1988	64 Kbps
EBONE (Integración)	1990	256 Kbps

Tabla 2.

Evolución de Ancho de Banda de las NREN Europeas 1982-1990.

Proyecto	AÑO	ANCHO DE BANDA
EBONE (Integración)	1991	2 Mbps
EUROPANET	1996	2 Mbps - 26 Mbps
TEN-34	1998	34 Mbps
TEN-155	2001	150 Mbps -622 Mbps

Tabla 2.1

Evolución de Ancho de Banda de las primeras redes avanzadas 1991-2001.

2.4. La red Avanzada Europea GEANT

GEANT se puso en marcha el 1 de noviembre del año 2000, fue financiada por DANTE y la Comisión Europea. Con una inversión de 200 millones de euros, fue sucesor del proyecto TEN-155 y tuvo como objetivo mejorar la infraestructura de las NREN europeas. En diciembre de 2001 GEANT se distribuyó a lo largo del continente europeo, con 27 países miembros y con conexiones en Gbps como se muestra en la tabla 3 y figura 8 [32- 34]

	Miembros de	GEANT 2001	
AT – AUSTRIA	EE- ESTONIA	IL- ISRAEL	PT- PORTUGAL
BE- BELGICA	ES- ESPAÑA	IT- ITALIA	RO- RUMANIA
BG- BULAGRIA	FR- FRANCIA	LT- LITHUANIA	SE- SUECIA
CH- SUIZA	GR- GRECIA	LU- LUXEMBURGO	SI- ESLOVENIA
CY- CHIPRE	HR- CROACIA	LV- LATVIA	SK-ESLOVAKIA
CZ- REP. CHECA	HU- HUNGRIA	NL- HOLANDA	UK- REINO UNIDO
DE- ALEMANIA	IE- IRLANDA	PL- POLONIA	

Tabla 3. Miembros de GEANT 2001.

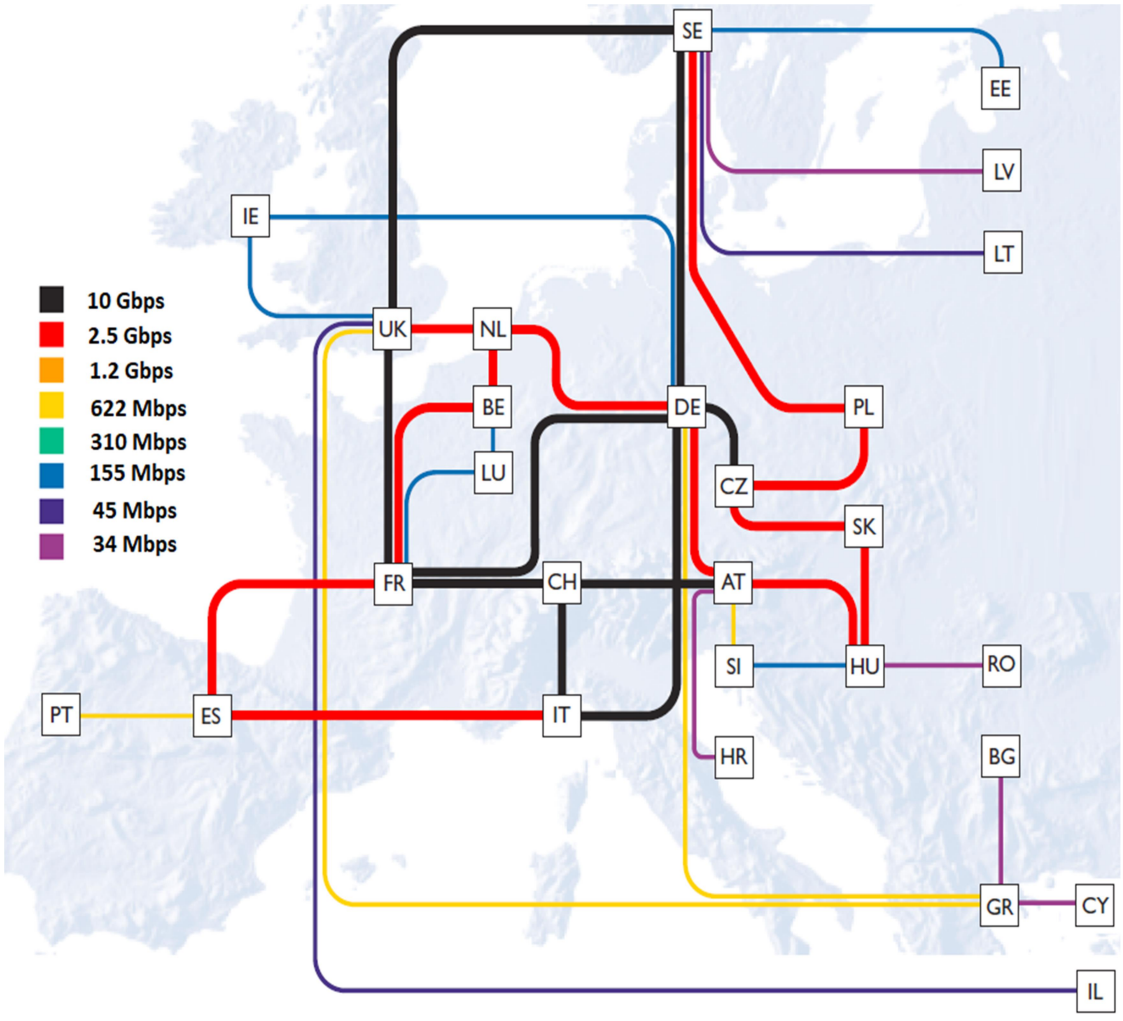


Figura 8. Topología de GEANT año 2001 [35].

Posteriormente en el marco del sexto programa 2002-2006 (6PM) de la Comisión Europea, se proporcionó un financiamiento al proyecto GEANT por 93 millones de euros a lo largo de 58 meses y para el año 2004 conectó a 33 países de Europa a través de sus redes nacionales y regionales como se muestra en la tabla 4 y figura 9 [36].

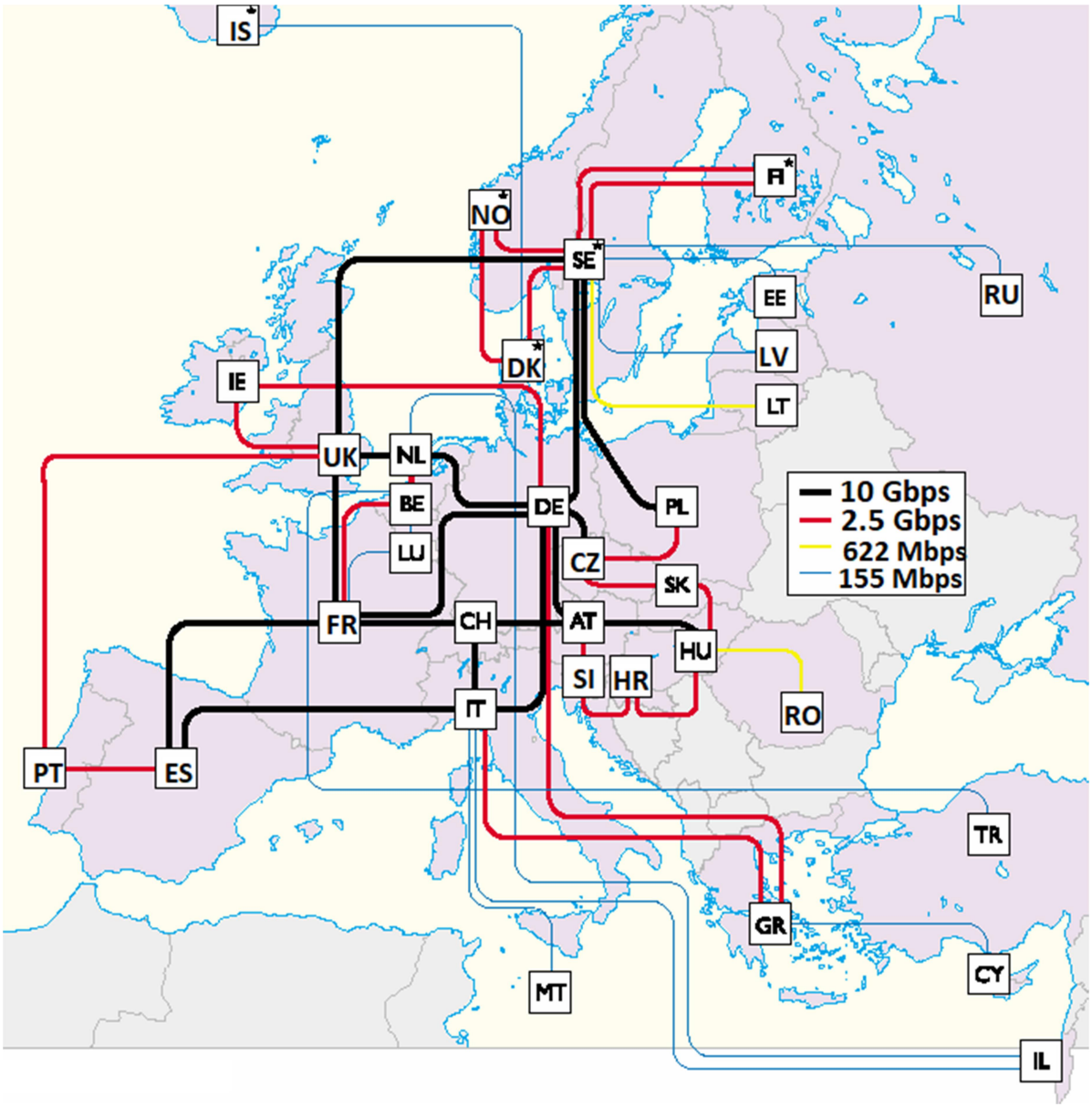


Figura 9. Expansión geográfica de GEANT abril 2004 [37].

AT- Austria	**CZ-** R. Checa	**ES-** España	**HR-** Croacia	**IS-** Islandia	**LV-** Latvia	**PL-** Polonia	**SE-** Suecia
BE- Bélgica	**DE-** Alemania	**FI-** Finlandia	**HU-** Hungria	**IT-** Italia	**MT-** Malta	**PT-** Portugal	**EL-** Eslovenia
CH- Suiza	**DK-** Dinamarca	**FR-** Francia	**IE-** Irlanda	**LT-** Lituania	**NL-** Holanda	**RO-** Rumania	**SK-** Eslovaquia
CY- Chipre	**ES-** Estonia	**GR-** Grecia	**IL-** Israel	**LU-** Luxemburgo	**NO-**Noruega	**RU-** Rusia	**TR-** Turquía **UK-** R. Unido

Tabla 4. Miembros de GEANT abril 2004.

2.4.1. Proyecto ALICE Y GEANT

Por otro lado, en junio de 2003 en el marco del programa "@lis" de la Comisión Europea y DANTE, crearon el proyecto ALICE (América Latina Conectada con Europa), con el propósito de conectar la red GEANT con la red avanzada de América Latina "CLARA". Con una inversión de 12.5 millones de Euros, se logró conectar las NREN de Argentina, Brasil, Chile, Panamá y México con la red GEANT, entre los PoP´s de Brasil y España con una conexión de 622 Mbps como se muestra en la figura 10 [38, 39, 40].

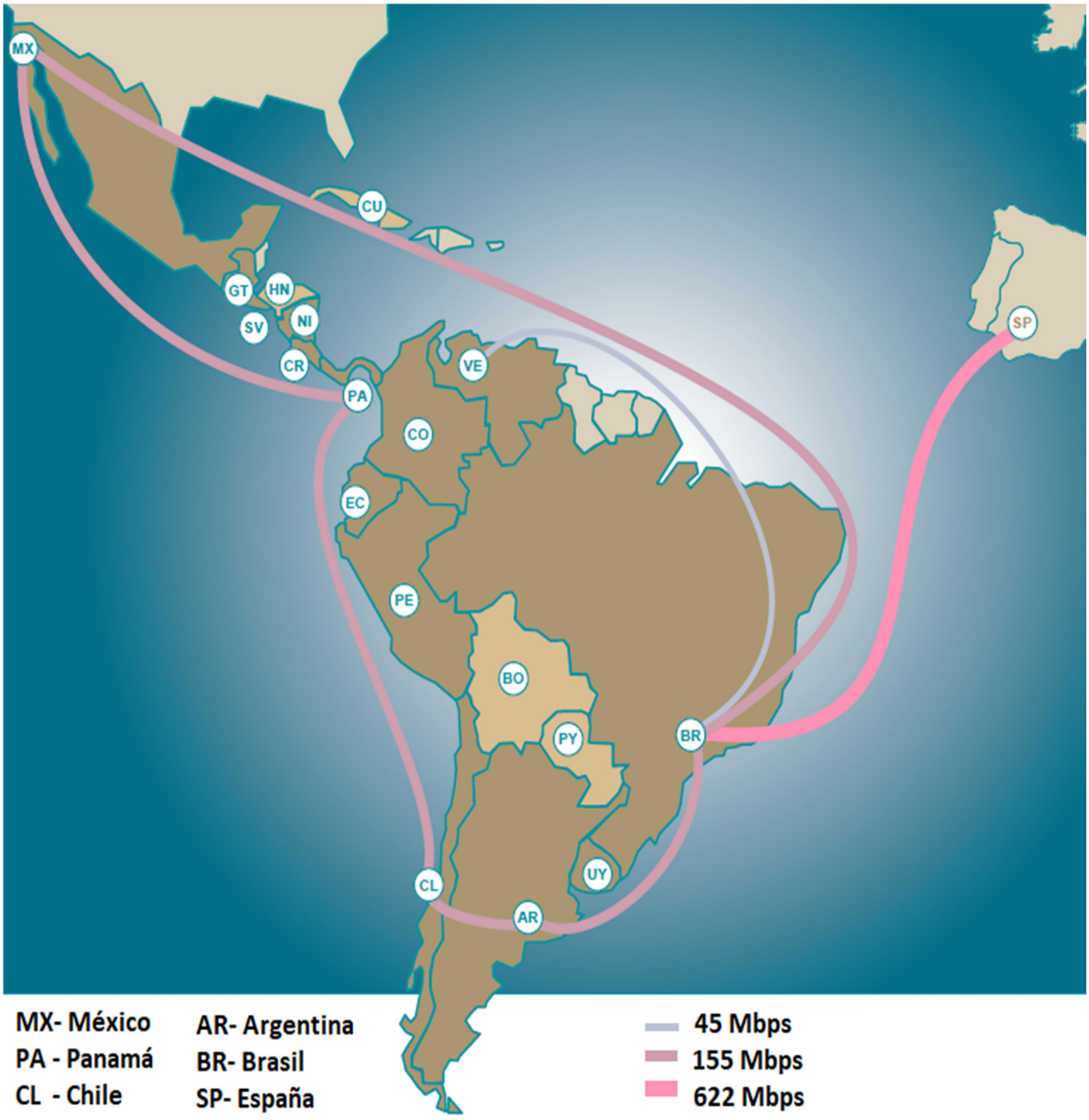


Figura 10. Proyecto ALICE (Conexión de GEANT y CLARA) Año 2004 [39].

Con el proyecto ALICE, se fomentó el desarrollo de proyectos entre investigadores europeos y latinoamericanos. El proyecto ALICE se terminó en abril del año 2006, sin embargo, gracias a los resultados favorables se extendió hasta marzo de 2008, teniendo participación de 12 países de América Latina como se muestra en la figura 11 [41, 42].

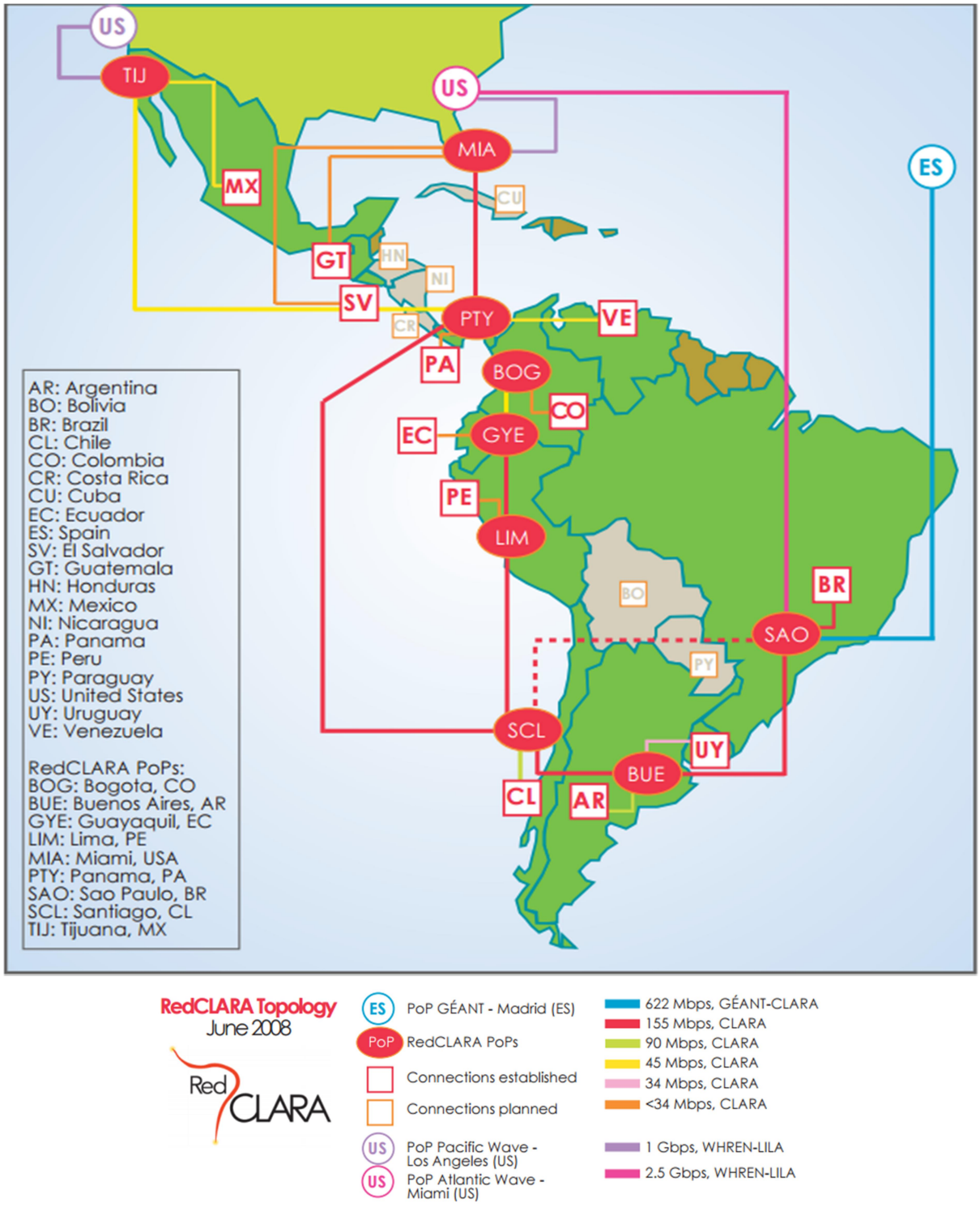


Figura 11. Proyecto ALICE Año 2008 Conexión de CLARA con GEANT [42].

2.4.2. GEANT 2

Fue anunciada en septiembre de 2004, como la segunda parte del proyecto GEANT, con una inversión de 200 millones de euros por parte de la Unión Europea, TERENA, DANTE y las NREN de Europa. Conectó a 34 países de Europa y a poco más de 30 millones de usuarios. Se distribuyó por más de 50,000 kilómetros en Europa, estableció vínculos con redes del Mediterráneo, Asia, África del Sur y América Latina, ofreció servicios avanzados como, IPv6, Premium IP, Multicast v4, v6 y ofreció velocidades de hasta 10 Gbps. Este proyecto concluyó a principios de 2009 y su última topología se muestra en la tabla 5 y figura 12 [43, 45, 46].

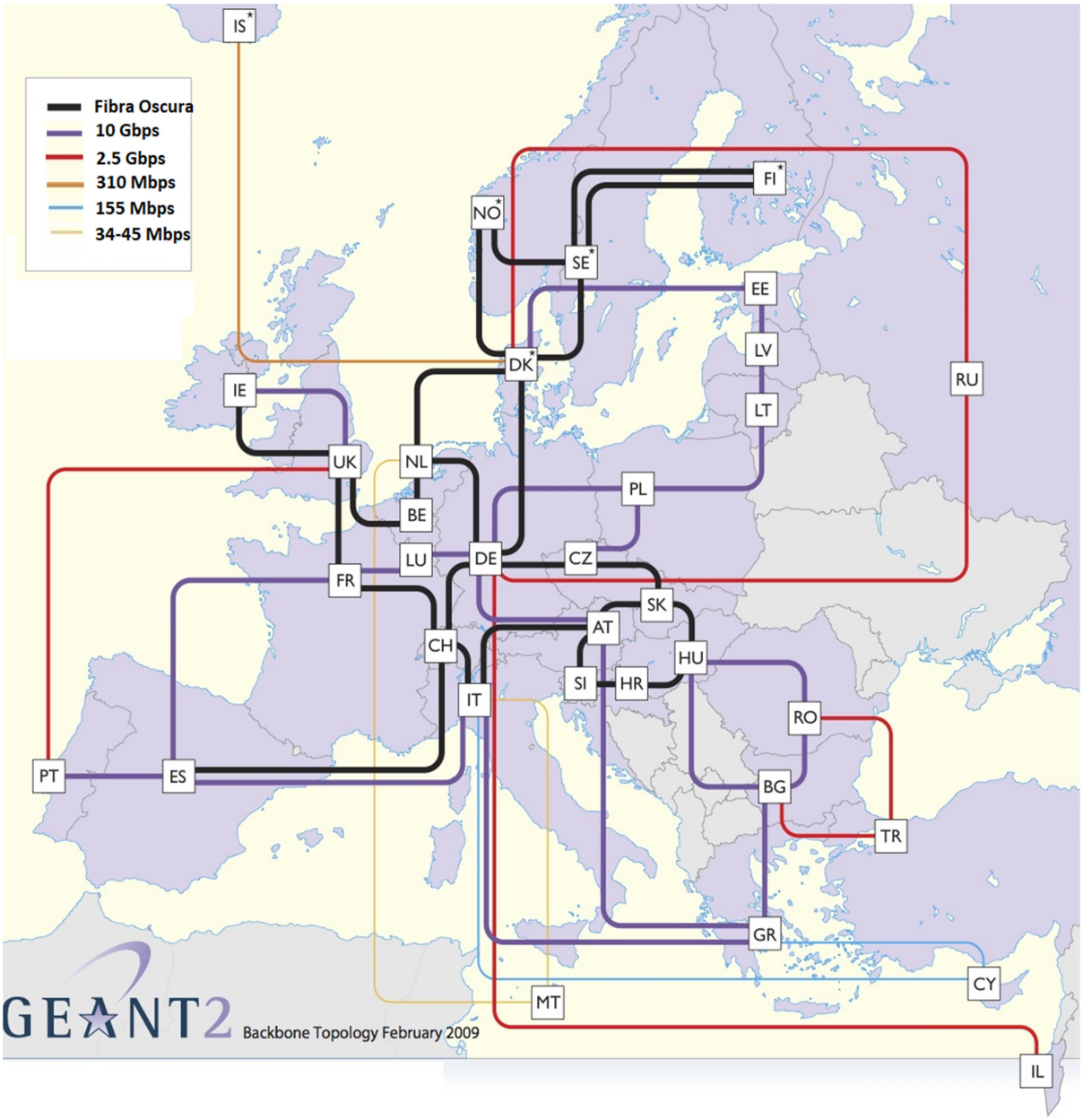


Figura 12. Topología de GEANT 2 Año 2009 [44].

AT- Austria	**CZ**- R. Checa	**FI**- Finlandia	**IE**- Irlanda	**LU**- Luxemburgo	**PL**- Polonia	**SI**- Eslovenia
BE- Bélgica	**DE**- Alemania	**FR**- Francia	**IL**- Israel	**LV**- Latvia	**PT**- Portugal	**SK**- Eslovaquia
BG Bulgaria	**DK**- Dinamarca	**GR**- Grecia	**IS**- Islandia	MT- Malta	**RO**- Rumania	**TR**- Turquía
CH- Suiza	**EE**- Estonia	**HR**- Croacia	**IT**- Italia	**NL**- Holanda	**RU**- Rusia	**UK**- Reino Unido
CY- Chipre	**ES**- España	**HU**- Hungria	**LT**- Lituania	**NO**- Noruega	**SE**- Suecia	

Tabla 5. Miembros de GEANT 2 Febrero 2009.

En comparativa con la topología de GEANT del año 2001(figura 7) y 2009 (figura 11), encontramos que la infraestructura de fibra oscura (*Lit Fibre*) aumentó lo que contribuyó en teoría a ofrecer una capacidad de ancho de banda para altas velocidades de conectividad. [45,46]

2.4.3. Proyecto ALICE segunda fase

En el año 2008 inició la segunda etapa del proyecto ALICE con la colaboración y financiamiento de la Comisión Europea y la red CLARA, con una inversión de 12 millones de euros por parte de Europa y 6 millones por parte de las NREN de América Latina, el objetivo fue estimular y apoyar la investigación entre América Latina y Europa, por medio de una infraestructura de red óptica. El proyecto finalizó en 2013 y en 2014 la red CLARA se conectó con GEANT con velocidades de hasta de 5 Gbps, entre los PoP´s de Brasil y Reino Unido como se muestra en la tabla 6 y figura 13 [47].

MX- México	**CR**- Costa Rica	**UY**- Uruguay	**EC**- Ecuador
US - Estados Unidos	**PA**- Panamá	**BR**- Brasil	**PE**- Perú
GT- Guatemala	**VE**- Venezuela	**UK**- Reino Unido	**CL**- Chile
SV - Salvador	**CO**- Colombia		**AR**- Argentina

Tabla 6. Países Miembros de la red CLARA y Proyecto ALICE año 2014.

Figura 13. Topología de la red CLARA y su Conexión con GEANT. [48]

2.4.4. GEANT 3

La tercera generación de GEANT inició en el año 2009, con un periodo de cuatro años. Se describió como una red de vanguardia tecnológica y conectividad avanzada. Fue financiada por la Comisión Europea, 32 socios de las NREN europeas, DANTE y TERENA. Este proyecto distribuyó infraestructura de fibra oscura por 19 países, poco más del 50% del Backbone europeo como se muestra en la tabla 7 y figura 14 [49-53].

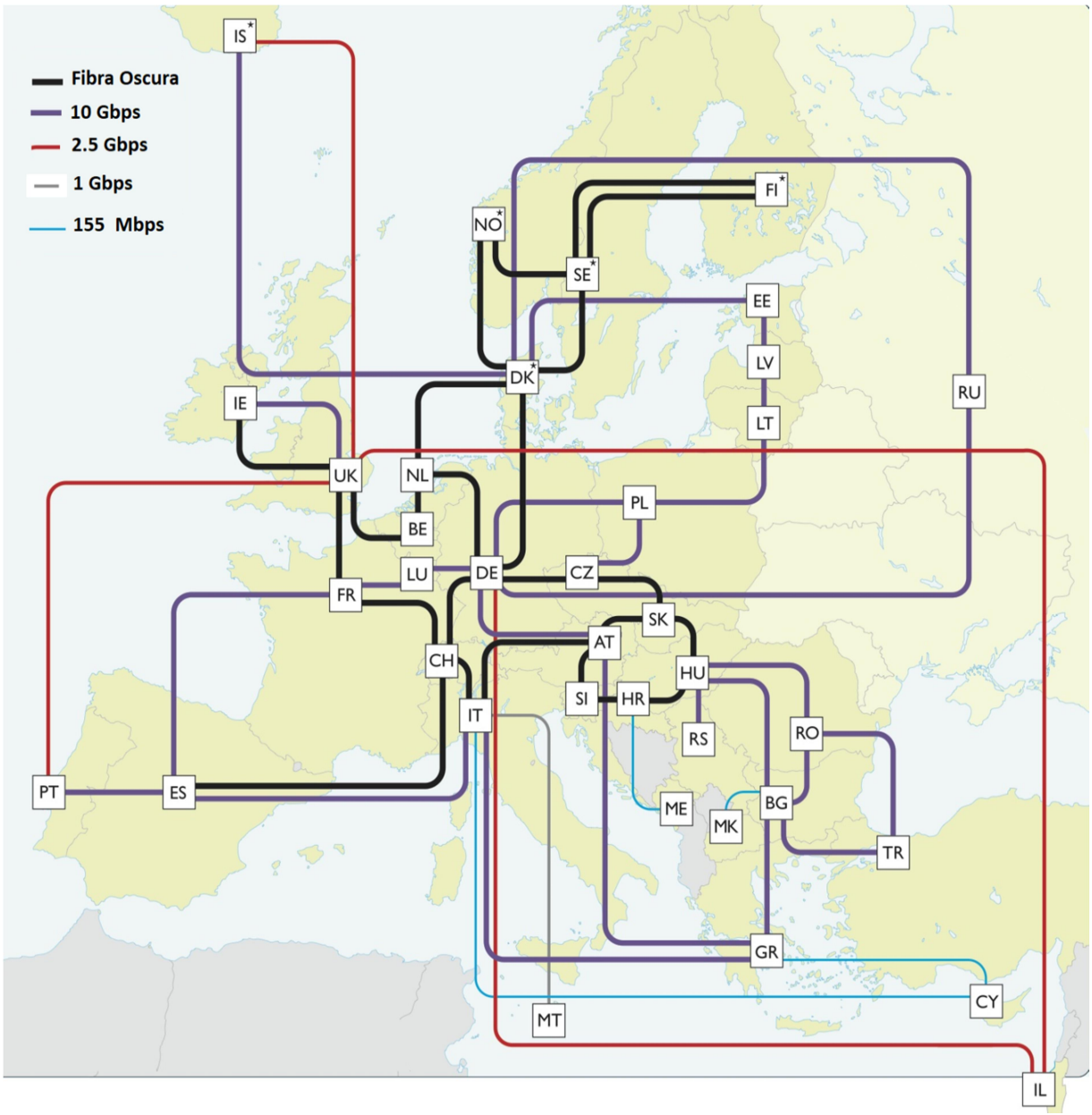


Figura 14. Expansión Geográfica de GEANT 3, Año 2010 [49].

GEANT 3 contó con la colaboración de 37 miembros, como se muestran en la tabla 4.

AT- Austria	**DE-** Alemania	**GR-** Grecia	**IT-** Italia	**MT-** Malta	**RS-** Serbia
BE- Bélgica	**DK-** Dinamarca	**HR-** Croacia	**LT-** Lituania	**NL-** Holanda	**RU-** Rusia
BG Bulgaria	**EE-** Estonia	**HU-** Hungria	**LU-** Luxemburgo	**NO-** Noruega	**SE-** Suecia
CH- Suiza	**ES-** España	**IE-** Irlanda	**LV-** Latvia	**PL-** Polonia	**SI-** Eslovenia
CY- Chipre	**FI-** Finlandia	**IL-** Israel	**ME-** Montenegro	**PT-** Portugal	**SK-** Eslovaquia
CZ- R. Checa	**FR-** Francia	**IS-** Islandia	**MK-** Macedonia	**RO-** Rumania	**TR-** Turquía **UK-**R. Unido

Tabla 7. Miembros de GEANT 3, año 2010 [49].

2.4.5. GEANT-3 PLUS

El proyecto GEANT 3 finalizó en el año 2013 para dar paso a GEANT 3 PLUS, el cual duro solo 2 años hasta 2015, el cual conectó a 41 países con potencial de velocidad mayores a 100 Gbps como se muestra en la figura 15 [54-56].

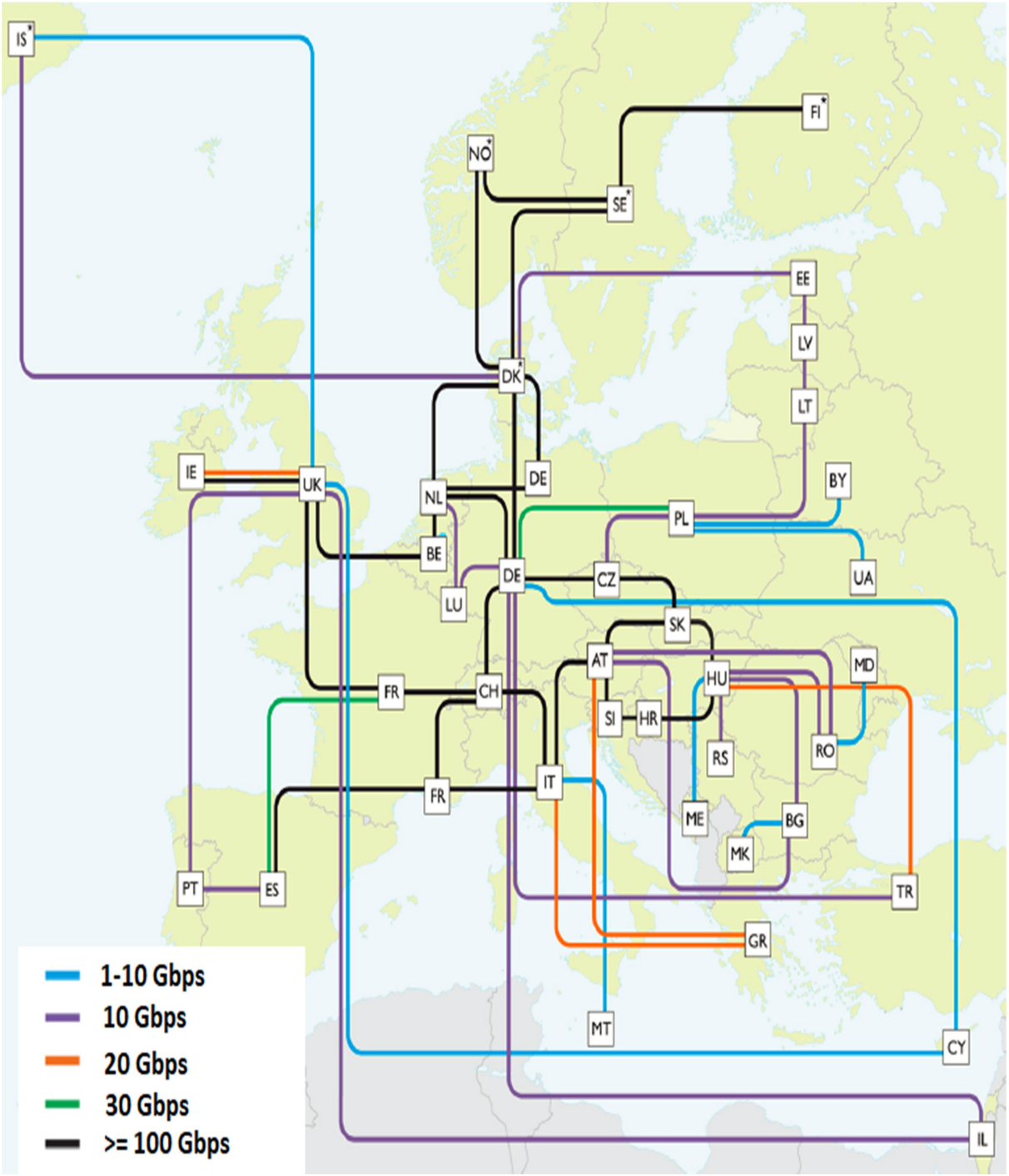


Figura 15. Topología de GEANT 3, año 2015 [54].

2.4.6. GEANT 4

La cuarta generación de GEANT está activa desde mayo de 2016 y trabaja en colaboración con las NREN europeas, NORDUNET y GEANT *Limited*, para conectar a poco más de 50 millones de usuarios y 10,000 instituciones de investigación y educación, en 41 países europeos, ofreciendo velocidades de hasta de 500 Gbps como se muestra en la figura 16 [57].

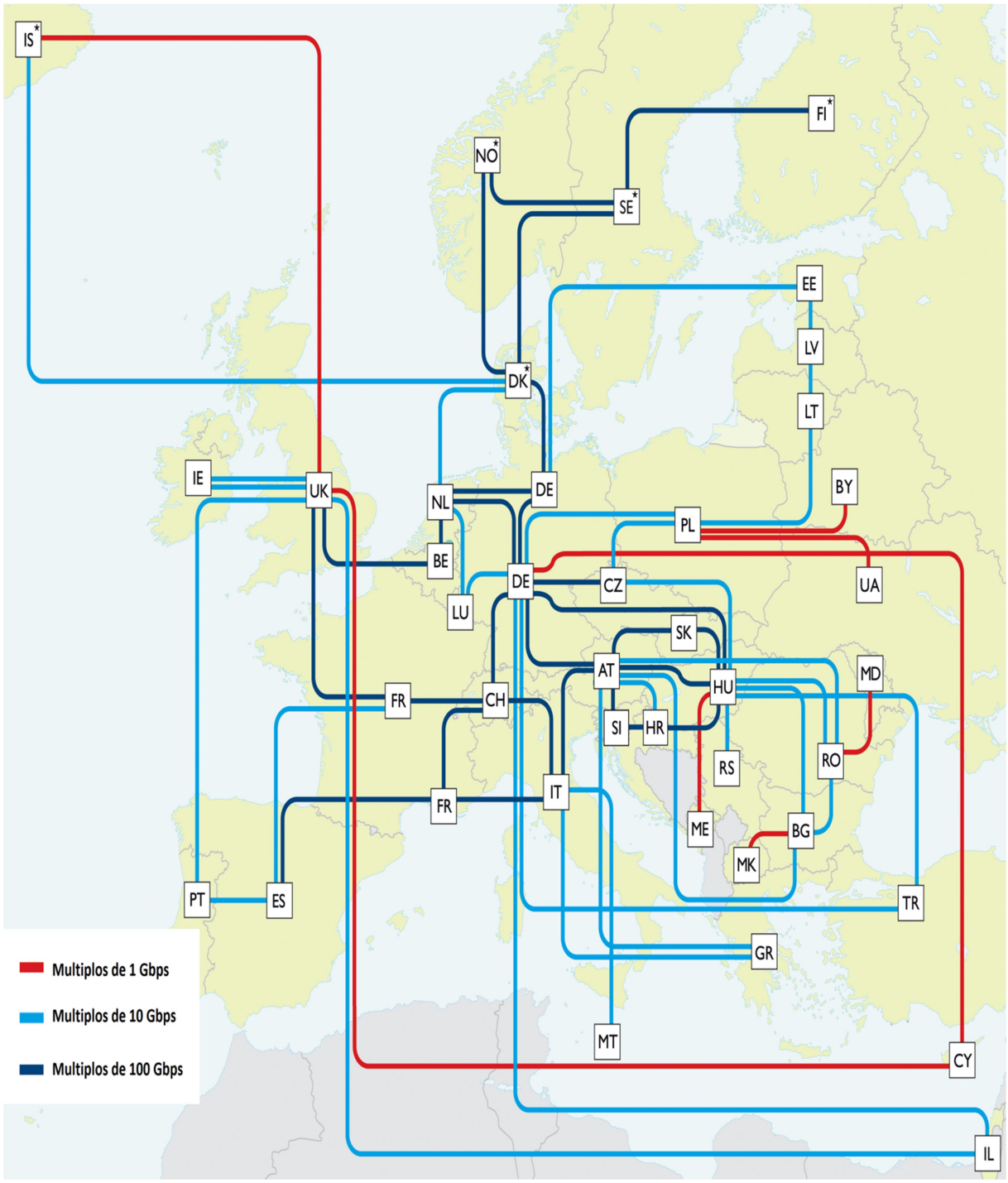


Figura 16. Topología de GEANT año 2017. [58]

GEANT, ha sido un proyecto de 15 años de éxito, que ha permitido el desarrollo de proyectos, investigaciones tecnológicas y científicas. Por ejemplo, el proyecto "EXPRESS" el cual, es un proyecto radioastronómico por parte de la Unión Europea que conecta un sistema de radiotelescopios distribuidos por China, Europa, Sudáfrica y Chile, que comparten información del monitoreo del clima espacial. Otro ejemplo, es el Gran Colisionador de Hadrones del CERN, el cual es considerado uno de los principales proyectos de la humanidad y exige una red con enlaces de alta velocidad para transmitir sin interrupciones grandes cantidades de datos entre centros de investigación, distribuidos en Europa y el resto del mundo. Aproximadamente su red transfiere 22 Petabits al año [59,60].

GEANT, evoluciona para poder soportar conectividad en Terabits por segundo (Tbps) aumentando su infraestructura de fibra en las NREN europeas y trabaja con equipos de última generación de conmutación avanzada para ofrecer una red de transporte Multi- Terabit que implican tecnologías de circuitos fotónicos en los equipos de comunicación basados en el flujo de fotones y el uso de protocolos de transporte DWDM (Dense *Wavelength Division Multiplexing*), GMPLS (*General Multiprocol Label Switching*) y protocolos OTN (*Optical transport Network*) [61]

La estructura del *Backbone* de GEANT, se divide en 2 capas, la capa de transmisión, basada en equipos de alta gama, con núcleos fotonicos, que integran cientos de componentes ópticos y permiten distribuir súper canales de hasta 500 Gbps y su capa de paquete la cual distribuye conexiones Ethernet punto a punto desde un punto de presencia de GEANT a una NREN o servicio privado, con velocidades de hasta 10 Gbps ofreciendo servicios IP plus.

Este trabajo analizara la topología del *Backbone* de GEANT de enero de 2017 a nivel de red, para probar conectividad y gestión de la capa de transmisión de GEANT, mediante la simulación, emulación, aplicación de protocolos de enrutamiento y de gestión.

Capítulo 3

Protocolos de Enrutamiento y de Gestión de red

3.1. La red de Internet y de las redes Avanzadas

La red Internet es un conjunto de redes descentralizadas que a través de estándares y protocolos permiten enviar y recibir información de un punto a otro sin importar la distancia, país o continente. Este proceso de comunicación se basa en el modelo del protocolo TCP/IP que se puede interpretar en términos de capas y protocolos, como se muestra en la figura 17 [62,63].

Capa	Protocolo
Aplicación	HTTP, Telnet, SNMP, DNS
Transporte	UDP, TCP
Internet	OSPF, RIP, ICMP
Enlace de datos	ATM, X.25
Física	Ethernet, Token Ring, USB

Figura 17. Capas del modelo TCP/IP

Tal que en la capa de aplicación, por medio de software de aplicación se interactúa de forma gráfica con una computadora para enviar o recibir información de la red, utilizando protocolos como HTTP, DNS entre otros. Posteriormente en la capa de transporte se designan los puertos de comunicación para enviar la información en forma de datagramas y transmitirlos a la capa de red la cual se encarga de hacer direccionamiento y escoger la mejor ruta para la información dividiéndola en paquetes. Enseguida la capa de enlace de datos, se encarga de formatear la información en forma de datagramas IP para poder transmitirlos en forma de bits por el medio físico de la red. Este proceso se repite de manera inversa en la solicitud-respuesta de la información en la red. [63]

El funcionamiento de una red avanzada es similar al de la red de Internet, con la diferencia de ofrecer un ancho de banda de alta velocidad dedicado a las NREN, usando infraestructura de redes de tipo OTN y routers de *Backbone* de alta gama [64].

Entendido lo anterior y enfocándonos a la conectividad en la simulación y emulación de la red GEANT, analizaremos a detalle los protocolos de enrutamiento de la capa de red, para posteriormente realizar pruebas de gestión con SNMP.

3.2. Protocolos de Enrutamiento

La comunicación en una red, sea Internet o en una Red Avanzada, se realiza por medio de conexiones entre routers, los cuales tienen como función descubrir e interconectar redes remotas para determinar la mejor ruta para el envío de paquetes. Esta acción se le conoce como enrutamiento y se clasifica en enrutamiento estático y dinámico. El enrutamiento estático se configura de forma manual definiendo una ruta directa entre dos o más routers, conocida como punto a punto. Las ventajas de utilizar enrutamiento estático es un consumo mínimo de ancho de banda ya que, no se usa ningún ciclo de CPU, es fácil de implementar en redes pequeñas y brinda seguridad en la red, debido a que el enrutamiento se hace de manera directa. Sus desventajas son que la actualización en la red se realiza de manera manual y no es apto para redes de gran tamaño ya que, conforme crece la red la configuración y el enrutamiento se vuelve más complejo. El enrutamiento dinámico consiste principalmente en la detección de redes, que utilizan algoritmos para determinar la mejor ruta origen-destino, mediante la configuración de protocolos que usan métricas y estructuras de datos para crear adyacencias entre los routers de la red. Estos protocolos son llamados protocolos de enrutamiento y se describen como un conjunto de procesos que se diferencian por el tipo de algoritmo y métricas que usan para realizar convergencia en la red. [65,66].

3.2.1. Clasificación de protocolos de enrutamiento

Los protocolos de enrutamiento se clasifican según su operación, comportamiento y propósito. Se clasifican por su operación, si trabajan dentro o fuera de un sistema autónomo-*Autonomous System*, es decir, si son de tipo IGP (*Interior Gateway Protocol*) o EGP (*Exterior Gateway Protocol*), por su comportamiento, si utilizan algoritmos vector- distancia o estado enlace y por su propósito si son protocolos con clase o sin clase [65].

Algunos de los protocolos de enrutamiento son RIP (*Routing Information Protocol*), IGRP (*Interior Gateway Routing Protocol*), BGP (*Border Gateway Protocol*) EIGRP (*Enhanced Interior Gateway Routing Protocol*), OSPF (*Open Shortest Path First*) e IS-IS (*Intermediate System-to-Intermediate System*) de cisco [64,65]

Los protocolos de enrutamiento han evolucionado debido a las exigencias, necesidades y crecimiento de las redes. En este trabajo abordaremos a detalle los protocolos de enrutamiento RIP y OSPF, describiendo su origen, operación, funcionamiento e implementación.

3.3. Protocolo de enrutamiento RIP v1

El protocolo RIP v1 surgió a partir del antiguo protocolo *Gateway Information Protocol* (GWINFO) de XNS (Xerox Network Systems, escrito por C. Hedrick de la Universidad de Rutgers en junio de 1988. Es un protocolo que trabaja sobre el modelo TCP/IP con clase, utiliza el algoritmo de Bellman-Ford conocido como Vector- Distancia para calcular la mejor ruta en una red, su métrica se basa en el conteo de saltos en donde 15 es el número máximo y en caso de alcanzar 16 se considera una ruta infinita y los paquetes se desechan. Por otro lado, la convergencia en la red se realiza con la distribución de tablas de enrutamiento que se comparten entre los routers por medio de mensajes de solicitud y respuesta. Los mensajes de solicitud piden a todos los routers vecinos configurados que envíen su tabla de enrutamiento y los mensajes de respuesta envían al router solicitante la tabla de enrutamiento. La actualización de estos mensajes se realiza por difusión (255.255.255.255) cada 30 segundos en toda la red y si no existe respuesta de algún router vecino en 180 segundos se asume que el router se ha deshabilitado y la ruta es inalcanzable [65, 66, 67,68].

3.3.1. Formato de mensaje RIP v1

El formato del mensaje RIP v1, es de 512 bytes y su estructura se divide en dos partes: la cabecera y el cuerpo de mensaje. La cabecera a su vez se divide en tres campos, el campo de comando, que indica el tipo de mensaje, es decir, si es de solicitud equivale a 1 y 2 equivale a respuesta. El campo de versión, nos indica la versión del protocolo, por último el campo "debe ser cero" está destinado para ofrecer espacio para futuras expansiones del protocolo. La segunda parte del formato de mensaje de RIP corresponde a la entrada de ruta que consiste en 3 campos; el identificador de familia de direcciones, dirección IP y métrica. En el identificador de familia se le asigna un 2 para identificar el protocolo IP, en el campo de dirección IP se asigna la dirección IP de las rutas de redes, este campo tiene capacidad para ingresar hasta 25 rutas y por último el campo de métrica, designa el conteo de saltos entre 1 y 15 saltos, como se muestra en la figura 18 [67, 69,70].

Comando (1 o 2)	Versión (1 o 2)	Debe ser cero
Identificador de familia		Debe ser cero
Dirección IPv4		
Debe ser cero		
Debe ser cero		
Métrica		

Figura 18. Formato de mensaje RIP v1 de 512 Bytes.

RIP v1 tiene algunas desventajas como el no distinguir si trabaja fuera o dentro de un sistema autónomo, es decir si son IGP o EGP, es un protocolo con clase lo que hace que su convergencia sea lenta, también no cuenta con algún tipo de seguridad en el envío de mensajes, ocupa ancho de banda en la difusión de mensajes cada 30 segundos y los mensajes disponen de mucho espacio no utilizado. [61, 69,70]

3.3.2. RIP v2

En el año 1993 surgió la segunda versión del protocolo RIP con el objetivo de mejorar RIP v1. Diseñado para ser un protocolo de enrutamiento sin clase, admitió VLSM (*Variable Length Subnet Mask*) y CIDR (*Classless Inter-Domain Routing*). Sus actualizaciones se hacen por multidifusión (224.0.0.9) para ahorrar sobre carga en la red, también se agregó autenticación MD5 para las actualizaciones de enrutamiento y se añadieron las etiquetas de ruta [71].

3.3.3. Formato de mensaje RIP v2

El formato de mensaje de RIP v2 es de 512 Bytes contiene una estructura similar al del RIP v1, a excepción de 2 nuevos valores que corresponden al *Subnetting* y a las etiquetas de ruta como se muestra en la figura 19.

Comando (1 o 2)	Versión (1 o 2)	Sin uso
Identificador de familia		Etiqueta de ruta
Dirección IPv4		
Máscara		
Siguiente salto		
métrica		

Figura 19. Formato de mensaje RIP v2.

La cabecera del formato de mensaje de RIP v2 es similar al formato de mensaje de RIP v1, a excepción del campo de etiqueta de ruta el cual es un atributo asignado a la identificación interna de RIP para diferenciar entre rutas internas (IGP) y rutas externas (EGP). En la segunda parte del mensaje se agregó el campo de máscara y este es destinado para el contenido de la máscara de subred, si el campo se marca como cero, no existe ninguna mascara de subred. También, se agregó el campo del siguiente salto que proporciona la ruta que seguirá el mensaje, con el propósito eliminar los paquetes que se generaron a través de saltos adicionales en el sistema [68, 71].

3.3.4. Bucle de enrutamiento

En ambas versiones del protocolo RIP, existe un problema cuando dos o más routers tienen información errónea en las tablas de enrutamiento, sus actualizaciones periódicas llenan las tablas con rutas inexistentes o inhabilitadas generando un destino inalcanzable y haciendo que el mensaje se transmita continuamente dentro de una red sin alcanzar su destino. Generalmente a este fenómeno se le conoce como bucle de enrutamiento. [71]

Para evitar estos eventos existen diferentes tipos de soluciones como son:

- Temporizadores de Espera
- Horizonte dividido
- Envenenamiento de Ruta
- Actualización instantánea.

Los temporizadores de espera se usan para acelerar el proceso de convergencia y también para evitar que los mensajes de actualización regulares reinstalen de manera errónea una ruta que puede no ser válida, es decir, cuando se detecta una ruta inalcanzable el router activa el temporizador de espera para que en las actualizaciones periódicas no se actualicen en esa ruta. El Horizonte divido condiciona un router para que no publique una red a través de la interfaz de donde se actualizó. El envenenamiento de ruta se utiliza para marcar una ruta inalcanzable, en una actualización que se envía o los demás routers, es decir, se envenena a un destino con un valor 16 para que la métrica lo considere inalcanzable. El horizonte dividido con envenenamiento, establece que al enviar actualizaciones desde una interfaz determinada se debe designar como inalcanzable a cualquier otra red sobre la cual se obtuvo la información mediante dicha interfaz. [67, 71]

3.3.5. Diferencias entre RIP v1 y RIP v2

Las principales diferencias entre la versión 1 y 2 de RIP se muestran en la figura 20.

Versión de RIP	Vector Distancia	Clase	Etiquetas de Ruta	Soporte de CIDR	Soporte de VlSM	Autentificación
V1	**SI**	**SI**	**NO**	**NO**	**NO**	**NO**
V2	**SI**	**NO**	**SI**	**SI**	**SI**	**SI**

Figura 20. Diferencias entre RIP v1 y RIP v2.

3.3.6. RIPng

En el año 1997 surgió la versión de RIPng (*RIP- Next Generation*) dedicada para el protocolo IP v6, basada su estructura en las versiones anteriores RIP, utiliza el algoritmo vector distancia y la métrica de conteo de saltos por lo que esta versión aún sigue siendo inadecuada para redes de gran tamaño [72].

3.3.7. Formato de mensaje de RIPng

La estructura del formato de mensaje es similar a RIP v2, la cabecera del mensaje es la misma y el cuerpo del formato de mensaje cambia con base al protocolo IPV6 agregando el campo de prefijo en donde indica el destino de la dirección IPV6 de 128 bits como se muestra en la figura 21 [65, 72].

<table>
<tr><th>Comando</th><th>Versión</th><th>Debe ser cero</th></tr>
<tr><td colspan="3">Ruta de entrada de tabla 1 (20)</td></tr>
<tr><td colspan="3">Ruta de entrada de tabla N (20)</td></tr>
<tr><td colspan="3">Prefijo IPv6 (16)</td></tr>
<tr><td>Etiqueta de ruta</td><td>Prefijo</td><td>Métrica</td></tr>
</table>

Figura 21. Formato de mensaje RIPng.

El protocolo RIP en sus tres versiones a pesar de tener mejoras no fue opción para satisfacer el enrutamiento de redes de gran tamaño por lo que se desarrollaron protocolos de enrutamiento como OSPF (*Open Shortest Path First*) e IS-IS. (*Intermediate System-to-Intermediate System*) [65].

3.4. Protocolo de enrutamiento OSPF v1

El protocolo OSPF (*Open Shortest Path First*) es un protocolo de enrutamiento TCP/IP diseñado para trabajar dentro de un sistema autónomo, considerado como un protocolo *Link-State* -estado enlace, se desarrolló en el año 1987 por parte del grupo de trabajo de OSPF y del IETF (*Internet Engineering Task Force*) y dos años más tarde se publicó la primera versión OSPF, la cual tuvo dos implementaciones, la primera para ejecutarse en routers, y la otra para ejecutarse en estaciones de trabajo UNIX, esta versión no tuvo una implementación importante y es considerada como una versión experimental [73-75].

3.4.1. OSPF v2

En el año de 1991 surgió la segunda versión de OSPF por John Moy y su actualización fue el 2 de abril de 1998. Utiliza el algoritmo de Dijkstra para encontrar el camino más corto y una métrica basada en costo. Es un protocolo sin clase y con una distancia administrativa de 110, ofrece autentificación MD5 para seguridad en la red en donde solo un grupo de routers autentificados puede interactuar [73,76].

El protocolo OSPF agrupa los equipos de enrutamiento en Áreas para generar un sistema jerárquico, en donde un área es un grupo de routers que comparten la misma información de estado de enlace por medio de base de datos de tipo LSDB (*Link State Data Base*). Se puede implementar en área única y multi área. En el área única los routers se encuentran en el área Backbone y en multi-área se divide el sistema autónomo en varias áreas de manera jerárquica, en donde todas las áreas se conectan al área cero. La ventaja de segmentar un sistema autónomo en áreas, es disminuir información de enrutamiento y segmentar una red como se muestra en la figura 22 [73,76].

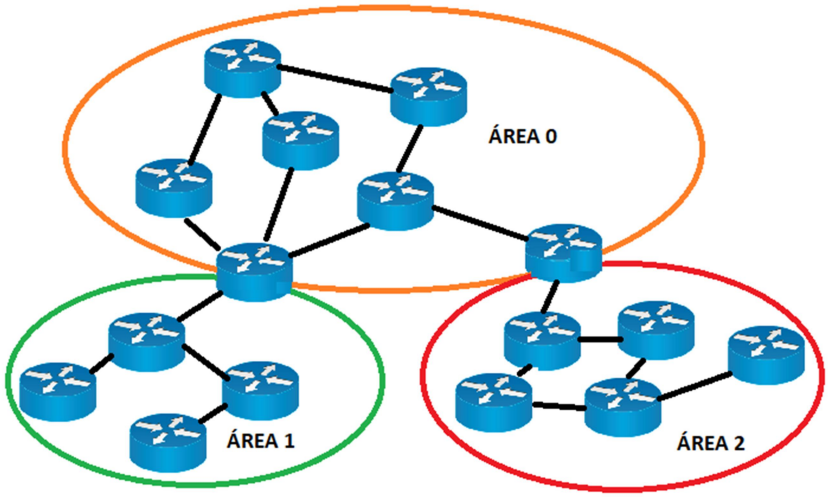


Figura 22. Agrupamiento lógico de Áreas en OSPF.

La implementación de multitarea OSPF en un sistema autónomo de gran tamaño, permite dividirlo en áreas más pequeñas generando un enrutamiento jerárquico que facilita la administración y procesamiento de las bases de datos ya que las direcciones de red se pueden resumirse entre áreas, reduciendo el procesamiento en los routers y disminuyendo la frecuencia de cálculos del algoritmo SPF. [73, 76]

3.4.2. Clasificación de routers OSPF

Cuando una red se divide en áreas, los routers se clasifican en ciertas categorías que definen su funcionamiento como a continuación se muestran:

Internal Routers (IR): Son routers internos conectados a la misma área y ejecutan el mismo algoritmo.

Area Border Routers (ABR): Son routers que se conectan y unen a diferentes áreas, estos almacenan y ejecutan diferentes algoritmos para cada área.

Backbone Routers (BR): Son routers que contienen una interfaz conectada al área de *Backbone*.

AS Border routers (ASBR): Son routers que interactúan con otros routers que pertenecen a otro sistema autónomo (AS). Ver figura 23. [73,76]

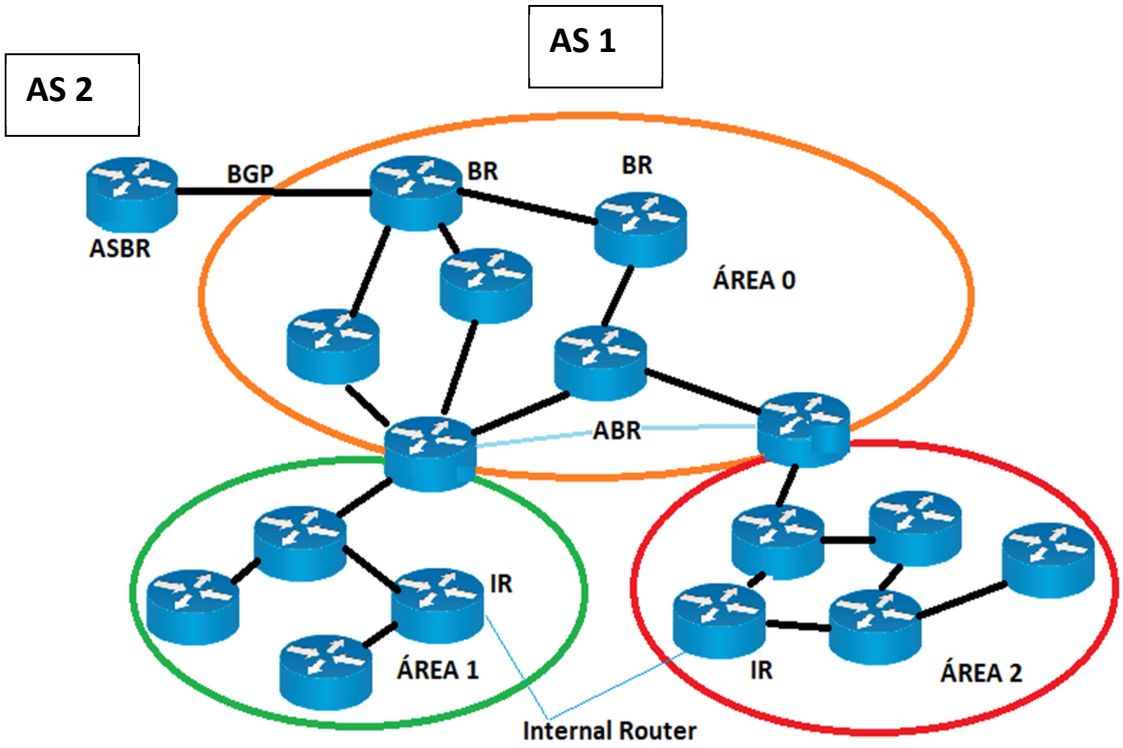


Figura 23. Tipos de Routers en OSPF.

3.4.3. Router DR y BDR

Otra característica importante que tiene OSPF, es reducir tráfico y saturación en la red designando un router DR (*Designated Router*), el cual representa un punto de recepción y distribución de los LSA (*Link State Advertisement*) de todos los routers, estos son avisos de las actualizaciones del estado de enlace de los routers y se utilizan para informar algún cambio en la red, como se muestra en la figura 24 [73].

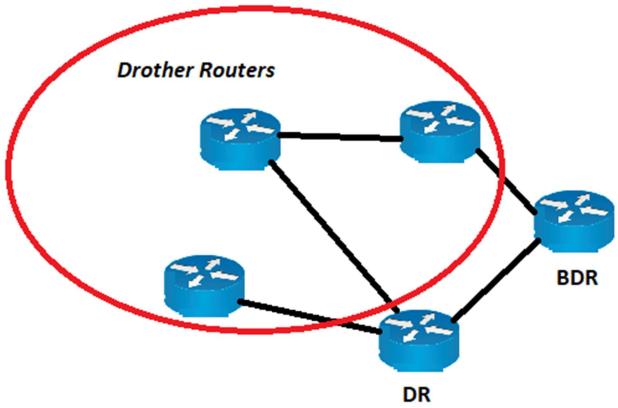


Figura 24. Router DR, BDR en un AS.

Los Router DR se puede asignar de manera automática o de manera manual. En el modo automático se configura el router con la dirección IP más alta configurada en *loopback* y en caso de no tener configurada una interface *loopback* se considera al router con mayor prioridad, si todos los routers tienen la misma prioridad se elige el router con la ID más alta, por otro lado también se puede asignar y un RDR de manera manual utilizando comandos [73].

El protocolo OSPF también designa un router BDR (*Backup Designated Router*), el cual es un remplazo del router DR y se elige mediante la segunda ID más alta, los router que no se configuran como DR ni como BDR se les llaman "*Drother*" [74, 76].

3.4.4. Tablas OSPF

Este protocolo utiliza mensajes para intercambiar información de enrutamiento y crear estructuras de datos que se procesan en el algoritmo de enrutamiento, estas estructuras de datos son las siguientes:

***Neighbor table*:** Es una tabla que contiene información de los routers vecinos. Los paquetes LSA son intercambiados en el momento de establecer adyacencias. Esta tabla se construye a partir los paquetes *Hello*.

***Topology table*:** Contiene todas las rutas posibles para alcanzar cualquier nodo de la red, si existe algún cambio en la red se genera un LSA nuevo.

Routing Table: Se genera a partir del resultado del algoritmo aplicado en la tabla topológica. En esta tabla se colocan las mejores rutas hacia cada destino. [70, 72,73]

3.4.5. Mensajes de tipo LSA

Como mencionamos anteriormente los LSA son mensajes que anuncian el estado de enlace de los routers de la red, estos se comparten cuando se forman adyacencias entre routers para formar los paquetes LSDB de cada router. Existen 5 tipos de mensajes LSA:

• ***LSA de Tipo 1***: Contiene una lista de todos los vínculos locales del router, el estado y el costo de sus enlaces. Son generados por todos los router y se hace multidifusión dentro del sistema autónomo.

• ***LSA de tipo 2*****:** Es generado por routers DR y contienen una lista de todos los routers conectados al router designado.

• ***LSA de Tipo 3*****:** Es generado por routers ABR y contienen una lista de todas las redes de destino dentro de un área. Los LSA de este tipo se envían para permitir la comunicación entre áreas.

• ***LSA de Tipo 4*****:** Son generados por un router ASBR y contienen una ruta a cualquier ASBR en el sistema OSPF. Son enviados desde un ASBR a su sistema autónomo para que los routers internos sepan cómo salir del sistema autónomo.

• ***LSA de tipo 5*****:** Son generados por un router ASBR y contienen rutas fuera del sistema autónomo, también pueden adoptar la forma de una ruta predeterminada para todas las redes fuera del sistema autónomo local, estos inundan a todas las áreas OSPF. [65,74, 76, 74]

3.4.6. Etapas de convergencia de OSPF

De acuerdo con lo que hemos explicado el protocolo OSPF designa áreas para una mejor convergencia y escalabilidad, asignando funciones específicas a los routers. Ahora explicaremos el funcionamiento y operación del protocolo OSPF. Este se puede resumir en 4 etapas como se muestran a continuación:

- *Neighbors relationship* - Creación de relación con router vecinos.
- *LSDB (Link-State Data base)* - Formación de base datos.
- *Exchange* - Intercambio de base de datos.
- *Route Calculation* - Calculo de rutas.

Para la etapa *Neighbors relationship,* se envía un mensaje *Hello* con el propósito de lograr reconocimiento de los routers vecinos y establecer una relación de adyacencia. Los mensajes "*Hello"*

se envían por *multicast*-multidifusión sobre la dirección de red 224.0.0.5, la cual se utiliza para enviar y recibir mensajes. Estos mensajes se usan para descubrir routers vecinos, pero también para intercambiar información de enrutamiento con el fin de mantener información acerca del estado de la red (envio de LSA). En la segunda etapa se forma la base de datos LSDB (*Link State Data Base*) en cada router, la cual es una representación completa de la red desde la perspectiva de cada router, esta se comparte con cada router. En la tercera etapa llamada *Exchange,* se realiza el intercambio de *LSDB* entre todos los routers que conforman la red. En la última etapa *Router Calculation*, se utiliza la información de las LSDB para calcular las rutas OSPF, es decir, se calculan las rutas con el algoritmo SPF creando un árbol SPF que posiciona a cada router en la raíz del árbol calculando las rutas con el costo más bajo hacia cada nodo, estas rutas se almacenan en la base de datos de OSPF y se crea una tabla de enrutamiento con el algoritmo SPF [65,74, 76,77].

3.4.7. Criterio de adyacencia en mensaje *Hello*

El proceso de adyacencia entre los routers debe cumplir ciertas condiciones en un paquete *Hello* los elementos listados a continuación deben de ser idénticos en cada router vecino [74-76].

- ID de área
- Tipo de área
- Máscara de subred
- Intervalo Hello
- Intervalo muerto
- Autenticación
- Tipo de red

3.4.8. Estados de adyacencia y convergencia de OSPF

El proceso de convergencia en OSPF se puede describir en 7 estados:

- Estado Down.
- Estado init
- Estado two-way
- Estado ExStar
- Estado Exchange
- Estado Loading
- Estado Full

Los tres primeros estados se consideran procesos para crear adyacencia entre routers y los demás consisten en la sincronización de las bases de datos OSPF. El estado *Down*, indica que no hay contacto con algún router vecino, el estado *init* consiste en él envió de un mensaje hacia un router vecino existiendo solo comunicación unidireccional, en el estado "*Two-way*" ya existe una comunicación bidireccional y se pasa al estado *ExtStar* en el que se crea la primera adyacencia, si la conexión es punto a punto se sincroniza de manera inmediata la base de datos pero si la conexión es Ethernet, se designa los routers DR y BDR. Cuando se logra adyacencia entre los routers se pasa al estado *Exchange,* el cual solicita información de sus LDSB y de sus estados de enlaces LSA, posteriormente

en el estado *Loading* se envía la información solicitada en *Exchange* y las rutas se procesan mediante el algoritmo SPF. Por último en el estado *full* se representa la convergencia de todos los routers de la red. [65,74, 76]

El cálculo de rutas se realiza con el algoritmo SPF, el cual utiliza las LSDB de los routers que previamente fueron compartidas en el proceso de adyacencia, con estas crea un árbol SPF el cual calcula las rutas más cortas para cada nodo de la red. El proceso se realiza con la métrica de costo y esta se basa en el ancho de banda de las interfaces. Para obtener la mejor ruta se suman todos los costos de interfaz de salida, comparando todas las rutas y se elige la que genere el menor costo. [65, 74,76]

3.4.9. Costo OSPF

En el proceso para calcular la mejor ruta, el algoritmo SPF utiliza el costo como métrica. Cuando el costo es menor, la ruta es la más apropiada para el algoritmo. Existen dos factores que determinan el costo OSPF, el ancho de banda de la interfaz y el ancho de banda de referencia, estos se relacionan en una ecuación en donde el ancho de banda predeterminado equivale a $10^{8.}$, como se muestra a continuación [65, 74,76].

Ecuación 1:

$$\boldsymbol{costo} = \frac{\text{ancho de banda de referencia}}{\text{ancho de banda de la interfaz}}$$

Ejemplo: Interfaz con ancho de banda de 10 Mbps. Se realiza la conversión del ancho de banda de la interfaz a "bps".

- Ancho de banda de referencia=100 Mbps=100, 000,000 bps
- Ancho de banda de la interfaz=10 Mbps= 10,000,000 bps

Ecuación 2:

$$\textbf{Costo= 10} = \frac{100000000}{10000000}$$

3.4.11. Tipos de Paquetes y Formato de mensaje OSPF

OSPF envía distintos tipos de paquetes en el proceso de convergencia de la red para crear adyacencias entre los routers vecinos como se muestra en la tabla 8.

Tipo	Nombre del paquete
1	Hello
2	DBD
3	LSR
4	LSU
5	LSACK

Tabla 8. Tipos de paquetes OSPF.

El paquete *Hello* se encarga de descubrir routers vecinos y establecer una adyacencia. El paquete DBD (*Data base Descriptor*) es un resumen de un LSDB que se usa para compararse con la LSDB local, los paquetes LSR se usan para pedir más información a los routers respecto a los DBD, el paquete LSU se utiliza para dar respuesta a esa petición por parte de LSR, por ultimo LSACK (*link-State Acknowledgement*), se encarga de que la multidifusión de las LSA sea confiable y reconoce a los demás tipos de paquetes [65, 74, 76].

3.4.10. Formato de mensaje del protocolo OSPF

Los mensajes del protocolo OSPF contienen información que se encapsula en un conjunto de capas y cada capa contiene un encabezado, el encabezado IP, el encabezado de trama de enlace de datos, el encabezado de OSPF y el encabezado de base de datos OSPF, como se muestra en la figura 25 [74, 76,78].

Encabezado de Trama de enlace de Datos	Encabezado de Paquete IP	Encabezado del paquete OSPF	Base de datos específicos del tipo de paquete OSPF

Figura 25. Encabezado de OSPF.

- ***Encabezado de la trama de Ethernet de enlace de datos:*** Identifica las direcciones MAC de multidifusión destino.
- ***Encabezado del paquete IP:*** Identifica el campo 89 del protocolo IPv4, como un paquete OSPF. Identifica direcciones OSPF por multidifusión 224.0.0.5.
- ***Encabezado del paquete OSPF:*** Identifica el tipo de paquete OSPF, la ID del router y la ID del área.
- ***Tipos de paquete OSPF:*** Identifica que tipo de información lleva el mensaje, si es un paquete Hello, DBD, LSR, LSU o LSACK.

El formato de mensaje OSPF tiene un tamaño de 32 bits como se muestra en la figura 26.

8 bits	8 bits	8 bits	8bits
Version	Type	Packet	Length
Router	ID		
Area	ID		
Checksum		AuType	
Authentication			
Authentication			
Packet	Data		

0x0000	Key ID	Authentication Data Length
Cryptographic	Sequence	Number

Figura 26. Formato de mensajes OSPF [71].

Cada campo del paquete OSPF identifica o describe una función como se muestra a continuación:

- ***Version*:** Indica que tipo de versión de OSPF utilizamos.
- ***Type***: Identifica que tipo de paquete se envía o recibe con opción del 1 al 5, como se mostró en la tabla 8.
- ***Packet Length***: Es el campo de la longitud del mensaje OSPF en octetos incluyendo el encabezado.
- ***Area ID***: identifica al área donde se originó el paquete.
- ***Checksum***: Es una comprobación de error del estándar IP del paquete.
- ***Autype**:* Es el tipo de autentificación que se utiliza:
    a. Sin autentificación.
    b. Autenticación con contraseña simple.
    c. autentificación MD5.
- ***Cryptographic Sequence Number*:** Es un valor numérico utilizado para evitar ataques de repetición.
- ***Key ID*:** Identifica el algoritmo de autenticación y la clave secreta utilizada.
- ***Authentication Data Length***: Especifica la longitud en octetos del resumen del mensaje añadido al final del paquete [65, 75, 76].

3.4.12. OSPF v3

El protocolo OSPFv3 surgió en el año 1999 creado por John Moy, Rob Coltun y Dennis Ferguson. La estructura es similar a la versión de OSPF V2 pero dedicada para el protocolo IPV6. Utiliza el algoritmo SPF para calcular las rutas en la red, las direcciones de red por ser IPv6 se les llama prefijo y las máscaras de red se les llama longitud de prefijo, los LSA se reemplazan por completo por bases de tipo TLV (*Type-Length-Value*) [83-89].

3.5. Gestión de Red

En esta sección describimos el protocolo de gestión SNMP para entender su propósito y que beneficios le ofrece a un administrador de red, ¿Pero qué significa la gestión en una red? de manera breve explicamos, que es la acción de planificar, monitorear, supervisar y controlar los elementos que conforman una red garantizando su óptimo funcionamiento. [80, 81]

3.5.1. Protocolo de gestión SNMP

SNMP (*Simple Network Management Protocol*), es un protocolo de administración y gestión de red que permite monitorear, controlar y configurar equipos a distancia en una red que opera con base en el modelo TCP/IP. Es un protocolo de la capa de aplicación y se divide en tres versiones SNMP v1 (Experimental), SNMP V2c (Comunidad) y SNMP v3 (Seguridad). El protocolo SNMP se estructura con tres elementos de gestión: [80,81]

Agente: Es un software que forma parte de un equipo gestionado el cual permite recibir y enviar información a la estación de gestión de red.

Dispositivos gestionados: equipos que conforman la red los cuales contienen un agente para enviar y recibir instrucciones para su control. Ejemplo; router, switch, computadora, impresora entre otros.

Estación de gestión de red: Es un Host configurado para controlar y monitorear dispositivos gestionados. Ver figura 27 [80,81, 84].

Figura 27. Dispositivos gestionados [84].

El proceso de gestión SNMP consiste en el envío y recepción de mensajes entre un agente y una estación de gestión de red, por medio del protocolo de transporte UDP usando los puertos de destino 161 y 162. Los mensajes son llamados Get, Set *y Trap.* El mensaje de tipo *Get* se divide en dos, la solicitud y respuesta (*Get-Request y Get-Response*) su función es monitorear y obtener información de los dispositivos gestionados a través de un host de gestión de red. El mensaje de tipo *Set* se encarga de modificar valores de los objetos en los agentes de los equipos gestionados. Por último el mensaje *Trap* es identificado como un mensaje de notificación o alerta por los dispositivos gestionados, su tarea es reportar eventos anormales de un equipo gestionado a la estación de gestión de red [80-82, 84].]

3.5.2. El árbol MIB

SNMP obtiene información de la red a través de un método denominado colección MIB, la cual es una colección de objetos organizados jerárquicamente en forma de árbol. Estos objetos tienen un OID (*Object Identifier*), que es un valor numérico que los identifica de manera única. Los objetos que están en el nivel superior del árbol pertenecen a diferentes organizaciones de estándares como ISO, CCITT e ISO-CCITT, mientras que el resto son asignadas por organizaciones o proveedores asociados e identificadores de objetos de Internet. Los proveedores pueden definir ramas privadas que incluyen objetos gestionados de sus productos. Para entender mejor cómo funciona el árbol MIB ponemos como ejemplo, encontrar el objeto "CiscoMg" por ello seguimos la ruta en el árbol MIB con la OID: "1.3.6.1.4.1.9.9" misma que representa los valores del sub árbol principal para el desarrollo de un MIB nuevo, como se muestra en la figura 28 [80, 81, 82,84].

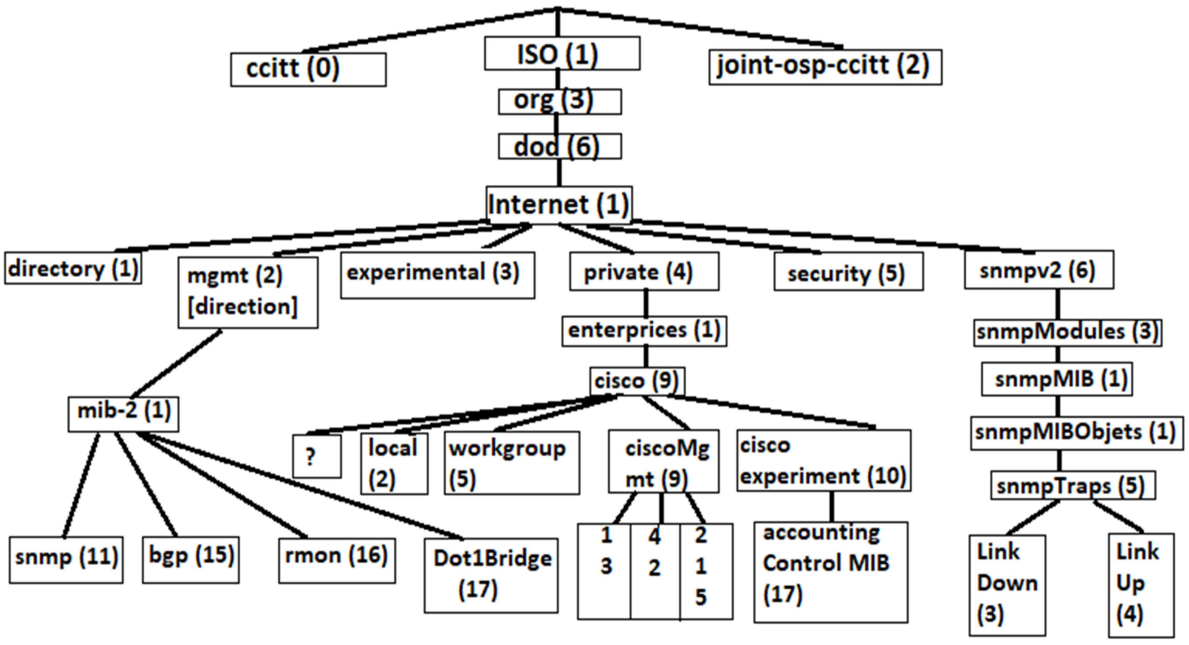


Figura 28. Árbol MIB.

3.5.3. SNMP v1

En el año de 1990 surgió la primera versión del protocolo SNMP, con una arquitectura solicitud Cliente –servidor, utilizando mensajes de tipo *Get, Set y Trap*. El mensaje *Get* se utilizó para obtener información del equipo gestionado por medio del agente, el mensaje *Set* permitió establecer valores de las instancias del objeto dentro de un agente, por último el mensaje Trap se caracterizó por ser un mensaje de alerta ante cualquier evento significativo en la red. La autentificación de los mensajes se basó en claves públicas que permitía que de los gestores realizaran peticiones de tipo *Get*-lectura y las claves privadas que permitían realizar peticiones de tipo *Set*-escritura [80,81, 84].

3.5.4. Formatos de mensajes SNMP v1

La estructura del mensaje SNMP v1 se divide en 2 partes, la cabecera de mensaje y la unidad de datos de protocolo (PDU) como se muestra en la figura 29 [80, 81]

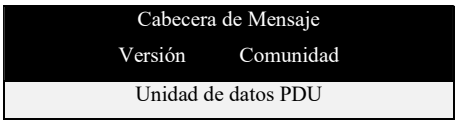


Figura 29. Formato de mensaje SNMP v1.

La cabecera de mensaje contiene 2 campos, especifica la versión de SNMP con un valor numerico y el nombre de la comunidad el cual define el entorno para la gestión, en la parte de la unidad de datos PDU contiene espacios específicos como se muestra en la figura 30.

PDU Type
Request Identifier
Error Status
Error Index
PDU Variable Bindings

Figura 30. Unidad PDU SNMP v1.

PDU type: especifica el tipo de PDU que se transmite en el formato de mensaje y se identifica con un valor numérico como se muestra en la figura 31 [81]

Valor-PDU	Tipo de PDU
0	*Get Request - PDU*
1	*Get Next Request - PDU*
2	*Response - PDU*
3	*Set Request PDU*
4	*Trap PDU SNMP v1*

Figura 31. Tipo de mensajes SNMP [81]

Request ID: Se genera en el dispositivo gestionado y se usa para distinguir el tipo de solicitud y asignarle la respuesta correspondiente.

Status error: Indica algún tipo de error en los mensajes por medio de un valor numérico, si es cero indica que no existe ningún error.

Error Index: Asocia un error con un objeto en particular y se usa para proporcionar información adicional al identificar la variable en la lista que causó el error.

Object Value 1, 2 y X: Están destinados para asociaciones de variables del árbol MIB [84]

3.5.5. Formato de mensaje TRAP PDU para SNMP v1

El formato Trap PDU consiste en 8 campos como se muestra en la figura 32.

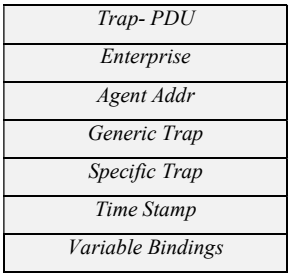


Figura 32. Formato trap PDU SNMP v1 [81]

Enterprise: Identifica el tipo de objeto gestionado.

Agent Address: Proporciona la dirección del objeto administrado para generar la *trap*.

Generic trap Type: Indica siete tipos de traps.

Valor	Tipo	Descripción
0	*ColdStart*	Reinicio de la entidad SNMP para modificar la configuración del agente.
1	*WarmStart*	Reinicio de la entidad SNMP para que no se pueda alterar la configuración del agente.
2	*LinkDown*	La entidad SNMP detecta una falla de enlace de la configuración del agente.
3	*LinkUp*	La entidad SNMP reconoce la habilitación de uno de los enlaces de configuración del agente.
4	*Authentication Failure*	Existe problema en la autentificación.
5	*EgpNeighborLoss*	Reconoce que un agente EGP se ha perdido y ya no existe.
6	*Enterprise Specific*	Reconoce un evento especifico de la empresa

Specific trap Code: Indica una serie de códigos de traps específicos para su implementación.

Time Stamp: Proporciona la cantidad de tiempo que ha transcurrido entre la última re inicialización de la red y el proceso de las traps.

Object Value: Son relacionados a la asociación de variables de objetos del árbol MIB [81, 84].

3.5.6. SNMP v2c

Es una mejora de la SNMP v1, se desarrolló en el año 1992 para solucionar problemas de seguridad de SNMP v1. Sin embargo, su seguridad se basó en cadenas de comunidad. Esta versión proporcionó nuevos aspectos de gestión, dando una mejor transferencia de datos y eficiencia de operación, se implementaron nuevos comandos como el *Getbuk request*, el cual recupera grandes bloques de datos y el *Inform Request* permite a la estación de gestión de red enviar *traps* de información a otra estación y luego recibir respuesta. Hubo mejoras en la estructura de las MIB y SMI (*Estructure of Management Information*) donde se especifica y documenta de manera más elaborada el manejo específico de información de gestión por medio de tablas [81,84-87].

3.5.7. Formato de mensaje SNMP v2

El formato de mensaje de SNMP v2 contiene una cabecera de mensaje y una parte para los PDU. La cabecera de mensaje contiene el número de versión como se muestra en la figura 33 [81, 85, 86].

<table>
<tr><td>Version</td><td colspan="2">Number = 2</td><td>Community</td></tr>
<tr><td>Model Number</td><td colspan="3">Quality of Service</td></tr>
<tr><td>Parameters</td><td colspan="3">Agent Identifier (12 bytes)</td></tr>
<tr><td colspan="4">Agent Number of Boots</td></tr>
<tr><td colspan="4">Agent Time Since Last Boots</td></tr>
<tr><td>User Length</td><td colspan="2" rowspan="2">User Name</td><td>Maximum Message Size</td></tr>
<tr><td></td><td>Authentication Digest Length</td></tr>
<tr><td colspan="4">Authentication Digest</td></tr>
<tr><td colspan="4">Context Selector</td></tr>
<tr><td colspan="4">PDU Control Fields</td></tr>
<tr><td colspan="4">PDU Variable Bindings</td></tr>
</table>

Figura 33. Formato de mensaje de SNMP v2.

Version Number*:* Campo que identifica la versión de SNMP. En este caso es el valor 2.
Model Number: Se establece con el valor 1 para identificar el tipo de modelo basado en usuario.
Quality of Service*:* Indica si la seguridad ha sido utilizada con la generación de un PDU.
Paramentes: Son un conjunto de parámetros que se utilizan para la seguridad con base en comunidad.
Agent ID: Se utiliza para identificar al agente que envió el mensaje.

***Agent Number of Boots**:* Es para informar el número de veces que se ha reiniciado el agente desde que le fue asignada su ID, con el fin de evitar un ataque de seguridad.

*Los campos siguientes fueron diseñados para implementar autentificación en SNMP v2, no obstante, fue hasta en SNMP v3 donde se implementaron.

User Length: Es el campo que almacena la longitud del valor del nombre de usuario.

***User Name**:* Es el nombre de Usuario que está mandando el mensaje.

***Authentication Digest Length**:* Es el campo destinado a la longitud de la autentificación.

***Authentication Digest**:* Es un valor de la autentificación para verificar la identidad y autenticidad del mensaje.

***Context Selector**:* Se utiliza para almacenar la información del mensaje para informar el contexto de la gestión.

***Message Body PDU*:** Destinados para la información del mensaje PDU.

3.5.8. Formato de PDU SNMP v2c

El formato PDU de SNMP v2, es similar al de SNMP v1 con un tamaño de 32 bits como se muestra en la figura 34 [81, 85, 86, 87].

PDU Type
Request Identifier
Non Repeaters
Max Repeaters
PDU Variable Bindings

Figura 34. Formato de PDU SNMP v2c.

3.5.9. Formato de mensaje Getbulk

El formato que se agregó en los mensajes SNMP v2 es el de GetBulk PDU, el cual consta de 5 campos. Ver figura 35 [81, 85, 86].

PDU Type
Request ID
Non repeaters
Max Repetitions
Object x Value x

Figura 35. Formato GetBulk PDU.

PDU Type: Identifica la PDU como una operación *GetBulk* con el valor 5.

Request ID: Asocia las solicitudes SNMP con las respuestas.

Non repeaters: Especifica el número de instancias de objeto en el campo de enlaces de variables que se deben recuperar más de una vez.

Max Repetitions: Define el número máximo de veces que se deben recuperar otras variables más allá de las especificadas por el campo.

Object 1 y 2: Sirve como campo para las variables de los objetos del árbol MIB.

Las estaciones de gestión en SNMPv1 y SNMPv2 utilizan comandos para obtener un valor de una o más variables. SNMP v2 Introduce dos nuevas características, el *GetBulk* para enviar grandes volúmenes de información de gestión y un comando "*Inform*" que se utiliza comunicar dos o más estaciones de gestión de red [81, 86, 87].

3.5.10. SNMP v3

Surgió como la versión posterior de SNMP v2, en abril de 1999 por el grupo de trabajo de J. Caso, con el único objetivo de brindar integridad y seguridad a los elementos de SNMP, mediante el modelo de seguridad conocido como USM (*User-based Security Model*) y el modelo de control de acceso llamado VACM (*View-based Access Control Model*). No obstante, la versión 3 aun ofrece seguridad basada en cadenas por comunidad [81, 84, 89].

La integridad y seguridad de SNMP v3 se manifiesta al verificar la integridad de los mensajes, por medio de la autentificación y verificar la identidad del usuario por medio del uso de claves cifradas en entre las entidades SNMP. Su estructura está basada en SNMP v2, sin embargo, la terminología para los elementos de gestión como los agentes o estación de gestión ahora son llamados "Entidades" que trabajan con un motor SNMP y los usuarios de servicio SNMP son llamados "identidades" [81, 89].

3.5.11. Motor SNMP v3

El motor SNMP en las entidades se compone de 4 elementos básicos:

1. Despachador (*the Dispatcher*)
2. Subsistema de procesamiento de mensajes (*Message Processing Subsystem*)
3. Subsistema de seguridad (*the Security Subsystem*)
4. Subsistema de control de acceso (*the Access Control Subsystem*)

El despachador tiene la función de enviar y recibir mensajes determinando la versión del protocolo de gestión en cada uno de los mensajes, una vez verificada la versión, se entrega el mensaje a las entidades. El subsistema de procesamiento de mensajes tiene como función enviar y extraer

información de los mensajes recibidos, este subsistema puede contener mensajes de distintos módulos de procesamiento. El subsistema de seguridad proporciona los servicios de autenticación y privacidad, reconociendo la autenticación para las versiones SNMP v1 y v2 basada en cadenas de comunidad y para SNMP v3, el modelo USM. Por último el subsistema de control de acceso tiene como función controlar el acceso a los objetos del árbol MIB [81, 88, 89].

3.5.12. Autentificación SNMP v3

La autentificación entre las entidades SNMP se realiza mediante algoritmos de tipo MD5 (*Message-Digest 5*) y SHA (*Secure Hash Algorithm*), en donde MD5 es un algoritmo de autentificación que por medio de una huella digital nos ayuda a corroborar la autentificación del mensaje, por otra parte SHA funciona de manera similar verificando los mensajes por medio de comparación de firmas digitales. Estos algoritmos se utilizan en SNMP v3 con el uso de claves entre las entidades al momento de configurar el protocolo de gestión. Las claves para el algoritmo MD5 contienen 16 octetos y para el algoritmo SHA contienen 20 octetos [86-89]

3.5.13. Privacidad SNMP v3

El módulo de privacidad SNMP v3 debe proporcionar seguridad contra la difusión de los mensajes SNMP y se realiza por medio de cifrado con criptografía DES (*Data Encryption Standard*) de 56 bits, 3DES de 168 bits y AES (*Advanced Encryption Standard*) de 128, 192 y 256 bits [81,88-90].

3.5.14. Aplicaciones SNMP v3

SNMP v3 se puede dividir en aplicaciones llamadas:

1. Command Generator: Genera: *Get, Getnext, Getbulk.* recibe solicitudes y procesa las respuestas
2. *Command responder*: Genera*: Get, Getnext, Getbulk* y establece solicitudes.
3. *Notification originator*: Genera *Traps* SNMP
4. *Notification receiver:* Recibe *Traps* y es implementada por la entidad gestora.
5. *Proxy forwarder*: Facilita el paso de mensajes SNMP entre entidades

Por lo anterior, una entidad SNMP v3 se compone de un motor SNMP dividido en 4 elementos y 5 aplicaciones como se muestra en la figura 36 [81, 84, 88,89].

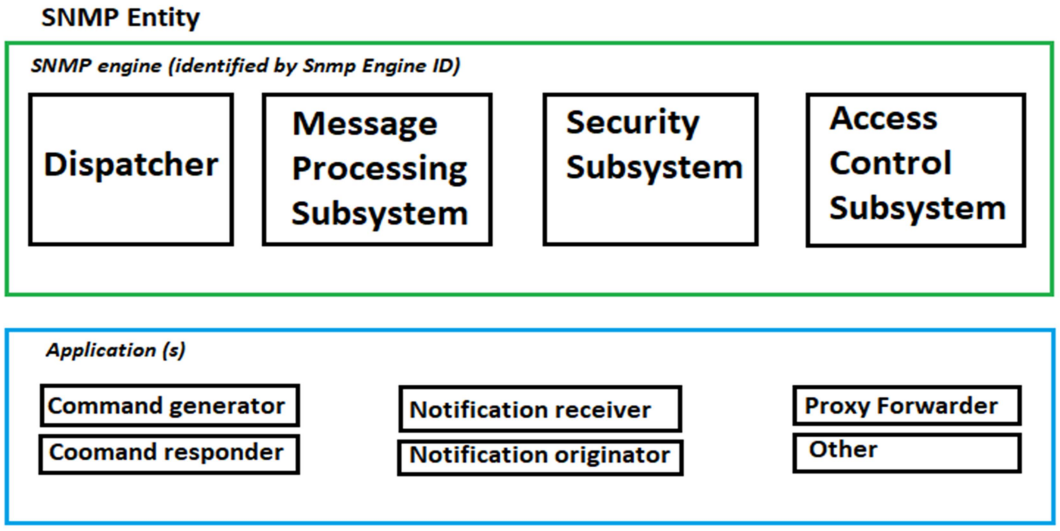


Figura 36. Composición de una Entidad SNMP v3 [84].

3.15.15. Formato de mensaje de SNMP v3

La estructura del mensaje de SNMP v3 adopta muchos componentes similares a la estructura del mensaje de SNMP v2, a excepción de los campos de seguridad y autentificación de SNMP v3 como se muestra en la figura 37 [81, 90].

Message Version	*Number =3*
Message	*Identifier*
Maximum	*Message Size*
Message Flags	*Message Security Model (Bytes 1 a 3)*
Message Security	*Parameters*
Context	*Engine ID*
Context	*Name*
PDU Control	*Fields*
PDU Variable	*Bindings*

Figura 37. Formato de mensaje de SNMP V3 [81].

***Message Version Number*:** Esta destino al número de versión del protocolo SNMP v3 en este caso sería el valor 3.

***Message Identifier*:** Es el identificador de mensaje y es utilizado para identificar un mensaje de SNMP v3 y hacerlo coincidir con su respuesta.

***Maximum Message Size**:* Es el tamaño máximo del mensaje que el remitente de este mensaje puede recibir.

Message Flags: Son un conjunto de indicadores que controlan el procesamiento del mensaje.

***Message Security Model*:** Es un valor que indica el modelo de seguridad para el mensaje por el modelo USM.

***Message Security Parameters**:* Son un conjunto de campos que contienen parámetros necesarios para implementar el modelo de seguridad utilizado en el mensaje.

***Context Engine ID*:** Se usa para identificar a la aplicación a la que se enviará el PDU para su procesamiento.

***Context Name*:** Es un identificador de objeto que especifica el contexto particular asociado a la PDU.

Scoped PDU: Contiene el PDU que se debe transmitir, junto con los parámetros que identifican a la entidad SNMP. Describe un conjunto de información de gestión accesible por una entidad particular. El campo puede ser encriptado o sin encriptar dependiendo del valor de *Priv Flag.*

Reserved: Espacio reservado para un uso futuro.

Reportable Flag: Indicador para cuando se establece en 1 una entidad para que envié un informe del tipo de PDU.

***Privacy Flag**:* es el indicador de privacidad y cuando establece un valor 1 indica que el cifrado se utilizó para proteger la privacidad del mensaje.

***Authentication Flag*:** Es el indicador de autentificación y se establece en 1, cuando indica que se utilizó la autentificación para proteger la autenticidad del mensaje. [81, 88-90]

Una vez visto el formato del mensaje de SNMP v3 observamos los complementos en los mensajes de gestión, en la seguridad y la autentificación, mientras que para el formato PDU se mantuvo sin cambios [81].

Capítulo 4

Simulación y Emulación de la Red GEANT

4. Simulación de la red Avanzada europea GEANT

Una vez conocido el origen y funcionamiento de las redes avanzadas, aplicaremos y pondremos a prueba los protocolos de enrutamiento y de gestión RIP, OSPF y SNMP, sobre la simulación de la red GEANT, utilizando el software de redes Cisco *Packet Tracer*, con base en la topología del *Backbone* europeo de enero de 2017 como se mostró en la figura 16.

4.1. Distribución de Interfaces de Red

La red GEANT conecta a 41 países entre los que destacan Alemania, Hungría, Austria y Reino Unido con el mayor número conexiones. Alemania conecta a 14 países del continente europeo con 2 PoP, Austria 8 países, Hungría 10 países y UK conecta a 6 países como se muestran en las tablas 8-11.

PoP 1 **Alemania (DE)**	**PoP 2** **Alemania 1 (DE1)**
Alemania 2 (DE1)	Alemania (DE)
Holanda (NL)	Rep. Checa (CZ)
Polonia (PL)	Chipre (CY)
Luxemburgo (LU)	Holanda (NL)
Suiza (CH)	Estonia (EE)
Israel (IL)	Dinamarca (DK)
Turquía (TR)	
Austria (AT)	
Hungría (HU)	

Tabla 8.

Países que se conectan a Alemania.

PoP **Hungría (HU)**
Austria (AT)
Montenegro (ME)
Croacia (HR)
Eslovaquia (SK)
Alemania (DE)
Rep. Checa (CZ)
Serbia (RS)
Rumania (RO)
Bulgaria (BG)
Turquía (TR)

Tabla 9.

Países que se conectan a Hungría.

PoP **Austria (AT)**
Italia (IT)
Alemania (DE)
Eslovaquia (SK)
Hungría (HU)
Bulgaria (BG)
Croacia (HR)
Grecia (GR)
Rumania (RO)

Tabla 10.

Países que se conectan a Austria.

PoP **Reino Unido (UK)**
Francia (FR)
Portugal (PT)
Bélgica (BE)
Chipre (CY)
Israel (IL)
Islandia (IS)

Tabla 11.

Países que se conectan a Reino unido.

4.1.1. Switch´s y routers en simulación de GEANT

El siguiente paso consiste en diseñar la topología de la red GEANT en el software de Packet Tracer, utilizando equipos genéricos para cada PoP de la red. Se utilizaran 42 routers, 42 switch´s y 42 Host (Computadoras), con conexiones de fibra óptica como se muestran en las figuras 38.

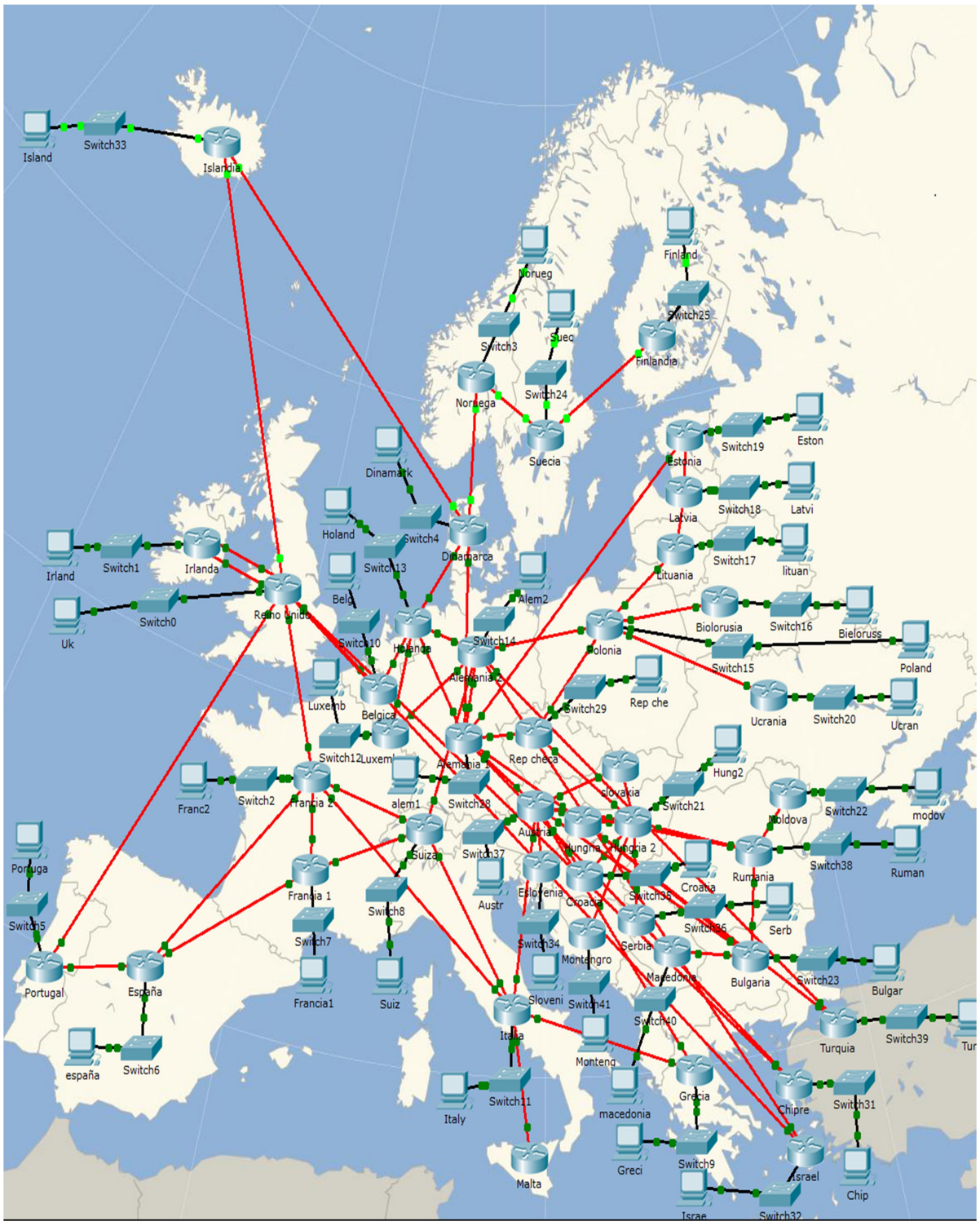


Figura 38. Diseño de la topología de red GEANT en simulación.

4.1.2. Configuración de redes IPv4

Una vez estructurada la red GEANT en el simulador propondremos las redes para cada país europeo utilizando el protocolo IPV4 con direcciones de red de clase C de tipo privadas como se muestra en la tabla 12.

	País	Red
1	Austria	192.168.10.0
2	Bélgica	192.168.39.0
3	Bulgaria	192.168.24.0
4	Suiza	192.168.7.0
5	Chipre	192.168.13.0
6	Rep. Checa	192.168.12.0
7	Alemania	192.168.4.0
8	Alemania 2	192.168.2.0
9	Dinamarca	192.168.15.0
10	Estonia	192.168.14.0
11	España	192.168.43.0
12	Finlandia	192.168.80.0
13	Francia	192.168.37.0
14	Francia 2	192.168.44.0
15	Grecia	192.168.32.0
16	Croacia	192.168.19.0
17	Hungría	192.168.11.0
18	Hungría 2	192.168.17.0
19	Irlanda	192.168.35.0
20	Israel	192.168.8.0
21	Islandia	192.168.40.0
22	Italia	192.168.31.0
23	Lituania	192.168.73.0
24	Luxemburgo	192.168.6.0
25	Letonia	192.168.77.0
26	Montenegro	192.168.18.0
27	Macedonia	192.168.53.0
28	Malta	192.168.46.0
29	Holanda	192.168.3.0
30	Noruega	192.168.78.0
31	Polonia	192.168.5.0
32	Portugal	192.168.38.0
33	Rumania	192.168.22.0
34	Serbia	192.168.21.0
35	Bielorrusia	192.168.74.0
36	Suecia	192.168.79.0
37	Eslovenia	192.168.33.0
38	Moldavia	192.168.67.0
39	Eslovaquia	192.168.20.0
40	Turquía	192.168.9.0
41	Reino Unido	192.168.36.0
42	Ucrania	192.168.75.0

Tabla 12. Redes IPv4 para los routers de la simulación de GEANT.

4.1.3. Interfaz de red en los routers de la simulación de GEANT

Teniendo en cuenta la asignación de la red en cada país europeo es necesario proponer las interfaces de red entre los routers para la conectividad entre los PoP de GEANT. Para el caso del PoP de Alemania, proponemos direcciones de clase B, tipo públicas como se muestran en las tablas 13-14.

Conexión	Red	Interface
Alemania	129.10.2.0	G1/0, G1/0
Holanda	130.10.2.0	G2/0, G2/0
Polonia	131.10.2.0	G3/0, G3/0
Luxemburgo	132.10.2.0	G4/0, G4/0
Suiza	133.10.2.0	G5/0, G5/0
Israel	134.10.2.0	G6/0, G6/0
Turquía	135.10.2.0	G7/0, G7/0

Tabla 13. PoP Alemania 2.

Conexión	Red	Interface
Austria	136.10.2.0	G2/0, G2/0
Hungría	137.10.2.0	G3/0, G3/0
Rep. Checa	138.10.2.0	G4/0, G4/0
Chipre	139.10.2.0	G5/0, G5/0
Holanda	140.10.2.0	G6/0, G6/0
Estonia	141.10.2.0	G7/0, G7/0
Dinamarca	142.10.2.0	G8/0, G8/0

Tabla 14. PoP Alemania.

Para Hungría, proponemos interfaces de red de clase B, de tipo públicas como se muestra en la tabla 15 y 16.

Conexión	Red	Interface
Alemania	137.10.2.0	G3, G3
Hungría 2	159.10.2.0	G8, G8
Bulgaria	157.10.2.0	G7, G7
Turquía	158.10.2.0	G9, G9

Tabla 15. Interfaces de red del PoP Hungría

Conexión	Red	Interface
Austria	150.10.2.0	G1,G1
Montenegro	151.10.2.0	G2, G2
Croacia	152.10.2.0	G3, G3
Eslovaquia	153.10.2.0	G4, G4
Rep. Checa	154.10.2.0	G5, G5
Serbia	155.10.2.0	G6, G6
Rumania	156.10.2.0	G7, G7
Hungría	159.10.2.0	G8, G8

Tabla 16. Interfaces de red del PoP Hungria2.

En el PoP de Austria, proponemos en su mayoría interfaces de red de clase A, de tipo públicas, como se muestra en la tabla 17.

Conexión	Red	Interface
Italia	10.10.2.0	G3, G3
Alemania	136.10.2.0	G2, G2
Eslovaquia	12.10.2.0	G5, G5
Hungría	150.10.2.0	G1, G1
Bulgaria	14.10.2.0	G4, G4
Croacia	15.10.2.0	G6, G6
Grecia	16.10.2.0	G7, G7
Rumania	17.10.2.0	G8, G8
Eslovenia	18.10.2.0	G9, G9

Tabla 17. Interfaces de red del PoP AUSTRIA.

En el PoP de Reino unido, propondremos interfaces de red de clase C, de tipo públicas como se muestra en la tabla 18.

Conexión	Red	Interface
Irlanda	192.10.2.0	G1, G1
Irlanda	193.10.2.0	G2, G2
Francia	194.10.2.0	G3, G3
Portugal	195.10.2.0	G4, G4
Bélgica	196.10.2.0	G5, G5
Chipre	197.10.2.0	G6, G6
Israel	198.10.2.0	G7, G7
Islandia	199.10.2.0	G8, G8

Tabla 18. Interfaces de red del Reino Unido.

El resto de interfaces de red entre los países europeos se propone como se muestra en la tabla 19.

Conexión	Red	Interface
Polonia- Lituania	90.10.2.0	G1, G1
Polonia- Bielorrusia	91.10.2.0	G2, G2
Polonia- Ucrania	92.10.2.0	G4, G4
Lituania- Letonia	93.10.2.0	G2, G2
Letonia – Estonia	94.10.2.0	G4, G4
Dinamarca- Noruega	95.10.2.0	G1, G1
Noruega- Suecia	96.10.2.0	G2, G2
Dinamarca- Holanda	98.10.2.0	G5, G5
Bélgica- Holanda	99.10.2.0	G7, G7
Holanda- Luxemburgo	100.10.2.0	G8, G8
Finlandia- Suecia	101.10.2.0	G1, G1
Islandia- Dinamarca	22.10.2.0	G3, G3
España- Portugal	40.10.2.0	G1, G1
Francia2- España	41.10.2.0	G2, G2
Francia2 - Francia	42.10.2.0	G1, G1
España – Francia	43.10.2.0	G4, G4
Francia- Suiza	44.10.2.0	G7, G7
Francia2- Suiza	45.10.2.0	G4, G4
Francia 2- Italia	46.10.2.0	G5, G5
Suiza- Italia	47.10.2.0	G1, G1
Malta- Italia	48.10.2.0	G2, G2
Italia- Grecia	49.10.2.0	G4, G4
Eslovenia- Grecia	50.10.2.0	G1, G1
Rep. Checa- Polonia	51.10.2.0	G6, G6
Macedonia- Bulgaria	52.10.2.0	G1, G1
Bulgaria-Rumania	53.10.2.0	G3, G3
Rumania- Moldava	54.10.2.0	G1, G1

Tabla 19. Interfaces de red del resto de los países europeos.

4.1.4. Configuración vía CLI y GUI de GEANT en simulación

Para la configuración de las interfaces el simulador nos ofrece dos modalidades, modo vía comando CLI (*Command- line Interface*) y modo grafico GUI (*Graphical User Interface*). Por vía comando es necesario abrir la consola del router e ingresar en modo súper usuario. Declaramos y configuramos la interfaz de red con el comando "*interface*" seguido del tipo y numero de interfaz, posteriormente le asignamos la dirección y mascara de red con el comando "*ip address*" como se muestra en la figura 39.

```
Router> en
Router# conf t
Enter configuration commands, one per line. End with CNTL/Z.
Router (config)# int GigabitEthernet1/0
Router (config-if)#ip add 129.10.2.3 255.255.0.0
Router (config-if)# no shut

Router (config-if)#
%LINK-5-CHANGED: Interface GigabitEthernet1/0, changed state to up

Router (config-if)#exit
Router (Config)#exit
Router#
%SYS-5-CONFIG I: Configured from console by console
```

Figura 39. Configuración de interfaz Gigabit Ethernet en router de Alemania.

Para la configuración grafica en la simulación abrimos la ventana de configuración del router y seleccionamos la pestaña "*Config*", posteriormente elegimos la opción de "*Interface*" la cual despliega las interfaces que contiene el router, elegimos la interface a configurar y llenamos los campos de "*IP Address*" y "*Subnet Mask*" con la dirección IP y su máscara de red, por ultimo habilitamos la interfaz en la opción "*Port Status*" como se muestra en la figura 40.

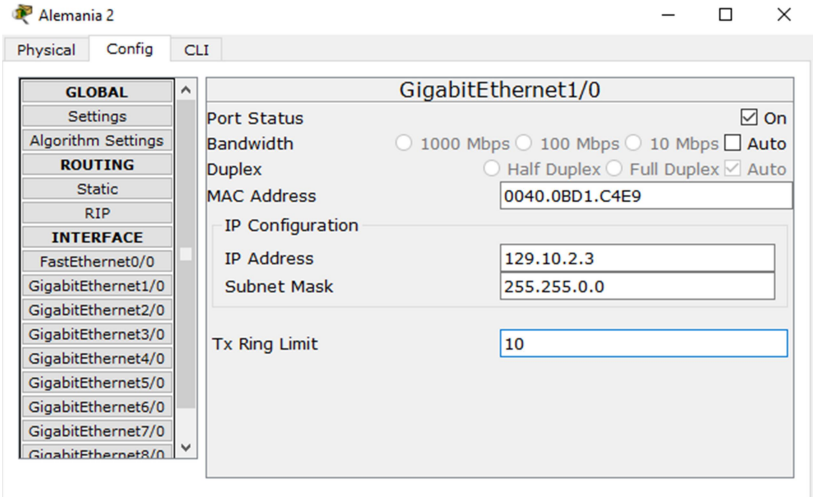


Figura 40. Configuración de Interfaz de router vía GUI.

4.1.5. Configuración de Protocolos de Enrutamiento (RIP v2)

Considerando la configuración de todas las interfaces de red en la simulación de GEANT y para lograr conectividad es necesario configurar el protocolo de enrutamiento. En modo CLI ingresamos a la consola del router e ingresamos en modo súper usuario, posteriormente ingresamos el comando "*router rip*" para habilitar el protocolo RIP, seguido del comando "*versión 2*". El siguiente paso es agregar todas las redes que se conectan al router con el comando "*network*" seguido de la dirección IP de cada red. Como ejemplo se muestran las configuraciones de los routers de Alemania y Alemania2. Ver figuras 41 y 42.

```
Router> en
Router# conf t
Enter configuration commands, one per line. End with CNTL/Z.
Router (config)#router rip
Router (config-router)#version 2
Router (config-router)#network 136.10.2.0
Router (config-router)#network 138.10.2.0
Router (config-router)#network 139.10.2.0
Router (config-router)#network 140.10.2.0
Router (config-router)#network 141.10.2.0
Router (config-router)#network 142.10.2.0
Router (config-router)#exit
```

Figura 41. Configuración del protocolo RIPv2 en router de Alemania.

```
Router> en
Router# conf t
Enter configuration commands, one per line. End with CNTL/Z.
Router (config)#router rip
Router (config-router)#version 2
Router (config-router)#network 129.10.2.0
Router (config-router)#network 130.10.2.0
Router (config-router)#network 131.10.2.0
Router (config-router)#network 132.10.2.0
Router (config-router)#network 133.10.2.0
Router (config-router)#network 134.10.2.0
Router (config-router)#network 135.10.2.0
Router (config-router)#exit
Router (config)#
```

Figura 42. Configuración del protocolo RIPv2 en router Alemania2.

Una vez configurados los routers de Alemania procederemos a configurar de la misma manera el protocolo RIPv2 en cada uno de los routers de la simulación de GEANT.

4.1.6. Configuración de OSPF

La siguiente etapa de nuestra simulación consiste en configurar el protocolo OSPF. Con el propósito de ver el comportamiento y funcionamiento en la simulación de un sistema autónomo dividido en áreas segmentaremos nuestra red de GEANT respecto a los PoP´s de países con más interfaces conectadas como lo son Alemania, Reino Unido y Hungria y dividiremos en 3 áreas como se muestra a continuación:

AREA 0	
1	DE Alemania
2	DE2 Alemania
3	CY Chipre
4	NL Holanda
5	LU Luxemburgo
6	IL Israel
7	TR Turquía
8	ES Estonia
9	LV Letonia
10	LT Lituania
11	DK Dinamarca
12	NO Noruega
13	SE Suecia
14	FI Finlandia

AREA 1	
1	UK Reino Unido
2	IE Irlanda
3	PT Portugal
4	FR Francia
5	FR2 Francia 2
6	IS Islandia
7	BE Bélgica
8	IT Italia
9	MT Malta
10	GR Grecia
11	CH Suiza
12	ES España

AREA 2	
1	HU Hungría
2	HU2 Hungría
3	RO Romania
4	MD Moldavia
5	BG Bulgaria
6	MK Macedonia
7	RS Serbia
8	ME Montenegro
9	HR Croacia
10	SK Eslovaquia
11	SI Eslovenia
12	AT Austria
13	CZ Republica checa
14	PL Polonia
15	BU Bielorrusia
16	UA Ucrania

4.1.7. Ancho de banda en la simulación de red GEANT

Dada la métrica de costo del protocolo OSPF que equivale a la relación del ancho de banda de referencia y el ancho de banda de la interfaz, es necesario configurar el ancho de banda de cada interfaz de la red GEANT, por lo que configuraremos a escala las velocidades de conexión, debido a que el software de simulación sólo nos permite configurar un ancho de banda de hasta 1Gbps como se muestra en la tabla 20.

Velocidades de la red GEANT	Velocidades en Packet Tracer
1 Gbps	10 Mbps
10 Gbps	100 Mbps
100 Gbps	1 Gbps

Tabla 20. Ancho de banda a escala de la red GEANT con ancho de banda de Packet Tracer.

4.1.8. Configuración de OSPF vía CLI en la simulación de GEANT

En esta etapa de la simulación configuramos cada router con el protocolo de enrutamiento OSPF. Utilizando el comando "*router ospf "* seguido del número de proceso para agregar las redes que se conectan al router con el comando "*network",* la dirección IP y la *"Wilcard",* que representa los bits apagados de la máscara de red. También necesitamos declarar el *área [x]"* a la que pertenece el router como se muestra en la figura 43.

```
Router> en
Router# conf t
Enter configuration commands, one per line. End with CNTL/Z.
Router (config)#router ospf 1
Router (config-router)#network 136.10.2.0 0.0.255.255 area 2
Router (config-router)#network 137.10.2.0 0.0.255.255 area 2
Router (config-router)#network 138.10.2.0 0.0.255.255 area 2
Router (config-router)#network 139.10.2.0 0.0.255.255 area 0
Router (config-router)#network 140.10.2.0 0.0.255.255 area 0
Router (config-router)#network 141.10.2.0 0.0.255.255 area 0
Router (config-router)#network 142.10.2.0 0.0.255.255 area 0
Router (config-router)#exit
Router (config)#
```

Figura 43. Configuración de router Alemania 2, área 0 con OSPF.

4.2. Configuración de protocolo SNMP en la simulación de GEANT

Una vez configurado OSPF, procederemos a la configuración del protocolo de gestión SNMP. Para esto necesitamos configurar el agente que está dentro del dispositivo gestionado (router). Como ejemplo elegimos el host del PoP de Alemania2 como la estación de gestión de red y para el dispositivo gestionado el router de la Alemania2 como se muestra en la figura 44.

Figura 44. Dispositivo gestionado y Estación de Gestión de red.

La configuración del protocolo SNMP en el simulador es necesario configurar los mensajes *Get* (Lectura "RO") y *Set* (Escritura "RW") además de declarar la comunidad para ambas operaciones. En este caso declaramos la comunidad *"geant*" para los mensajes de tipo *Get y Set*, con los comandos "*snmp-server community geant ro"* y *"snmp-server community geant rw"* como se muestra en la figura 45.

```
Router> en
Router# conf t
Enter configuration commands, one per line. End with CNTL/Z.
Router (config)# snmp-server community geant ro
%SNMP-5-WARMSTART: SNMP agent on host Router is undergoing a warm start
Router (config)# snmp-server community geant rw
Router (config)#exit
Router#
%SYS-5-CONFIG_I: Configured from console by console
```

Figura 45. Configuración del agente en el router de Alemania2

4.2.1. Configuración de MIB Browser en Packet Tracer

El software de simulación Packet Tracer nos ofrece la opción MIB Browser en los Host para poder simular nuestra estación de gestión de red y acceder al agente de los dispositivos gestionados en la simulación. Para su configuración de MIB Browser abriremos el host de Alemania2, seleccionaremos la opción de escritorio (*Desktop*) y la opción MIB como se muestran en las figuras 46 y 47.

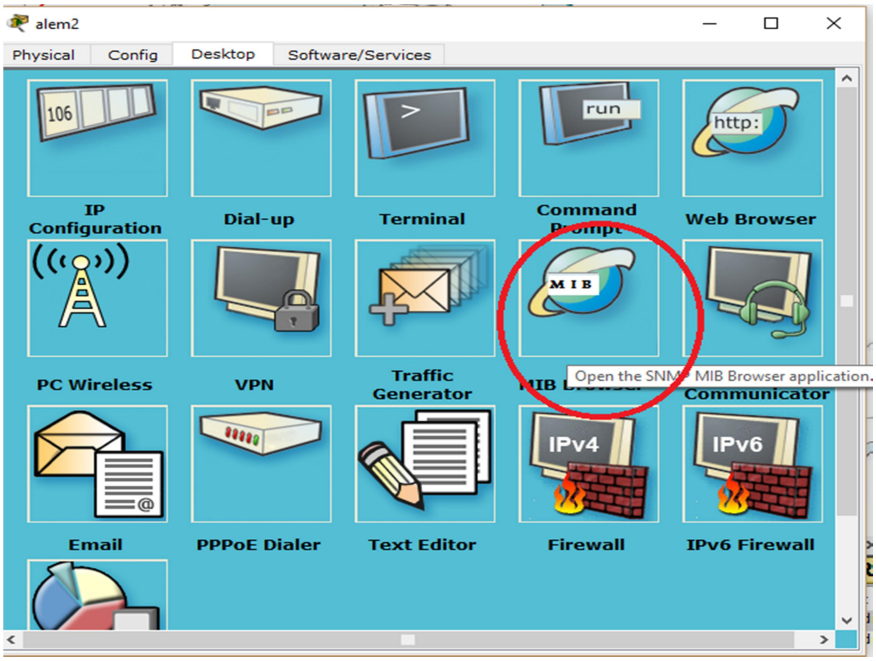


Figura 46. Escritorio de host Alemania2.

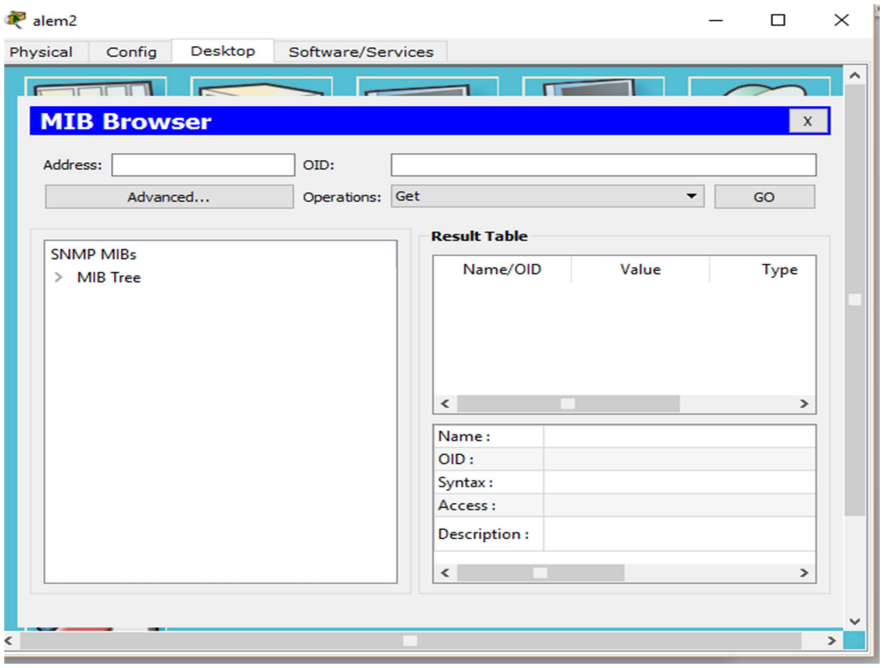


Figura 47. MIB Browser router Alemania2.

Posteriormente configuramos la *Gateway* en el campo (*Address*) de nuestro *host*, después en la opción *Advanced*, agregamos la comunidad "*geant*" para la operación *get y set* y por ultimo seleccionamos la v2 del protocolo SNMP, como se muestra en la figura 48.

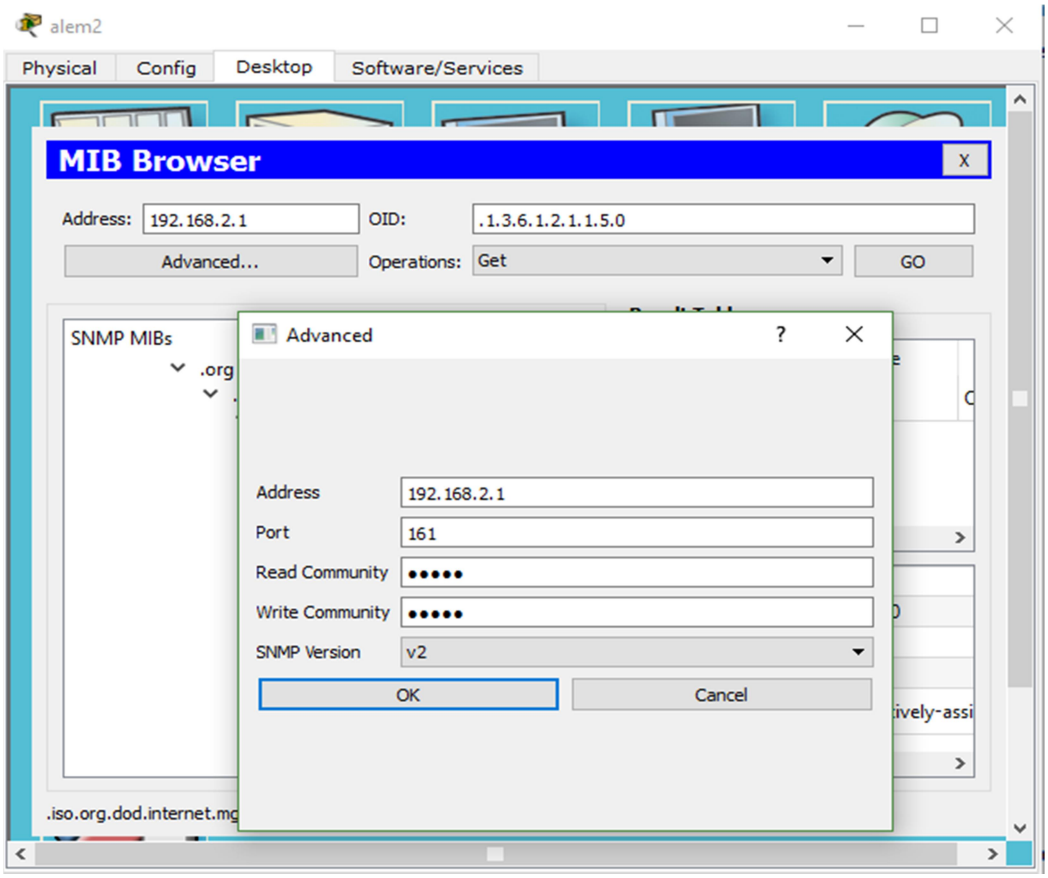


Figura 48. Configuración MIB Browser.

4.3. Emulación de la red avanzada GEANT

En esta sección realizaremos la configuración de la red GEANT en el emulador GNS3 aplicando el protocolo de enrutamiento OSPF y el de gestión SNMP, sin embargo, el funcionamiento del GNS3 es distinto al simulador Packet Tracer ya que necesitamos instalar IOS de routers y agregar máquinas virtuales. Para agregar los IOS de los routers abrimos la pestaña *edit* del programa GNS3 → *Preferences* → *Dynamips* → *IOS routers* → *New* → *New Image* → *Browser* → elegimos la imagen IOS del router en este caso utilizaremos el IOS del router *c7200-adventerprisek9-mz.150-1.M,* lo seleccionamos→ *Next* → *Next* → **y** configuramos los *Slots* o interfaces a utilizar como se muestra en las figuras 49-52.

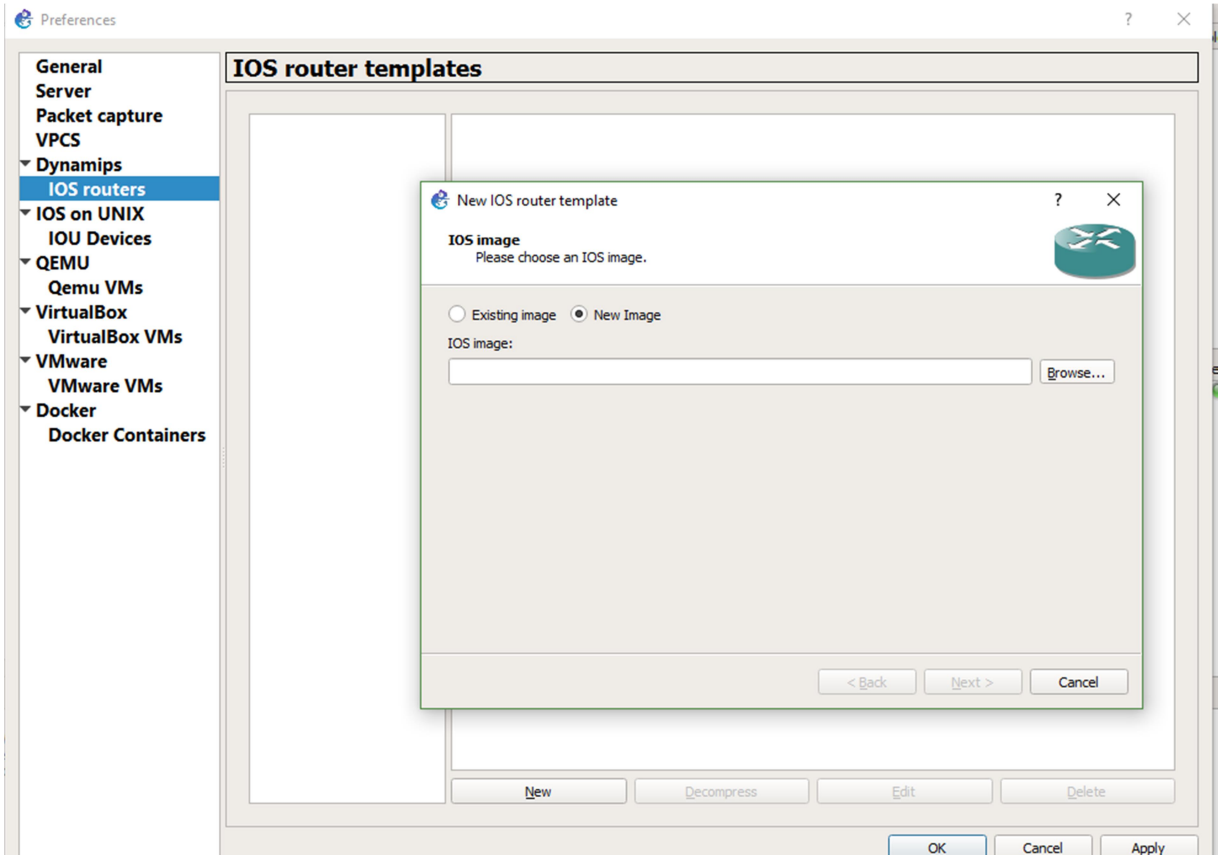


Figura 49. Selección de IOS del Router.

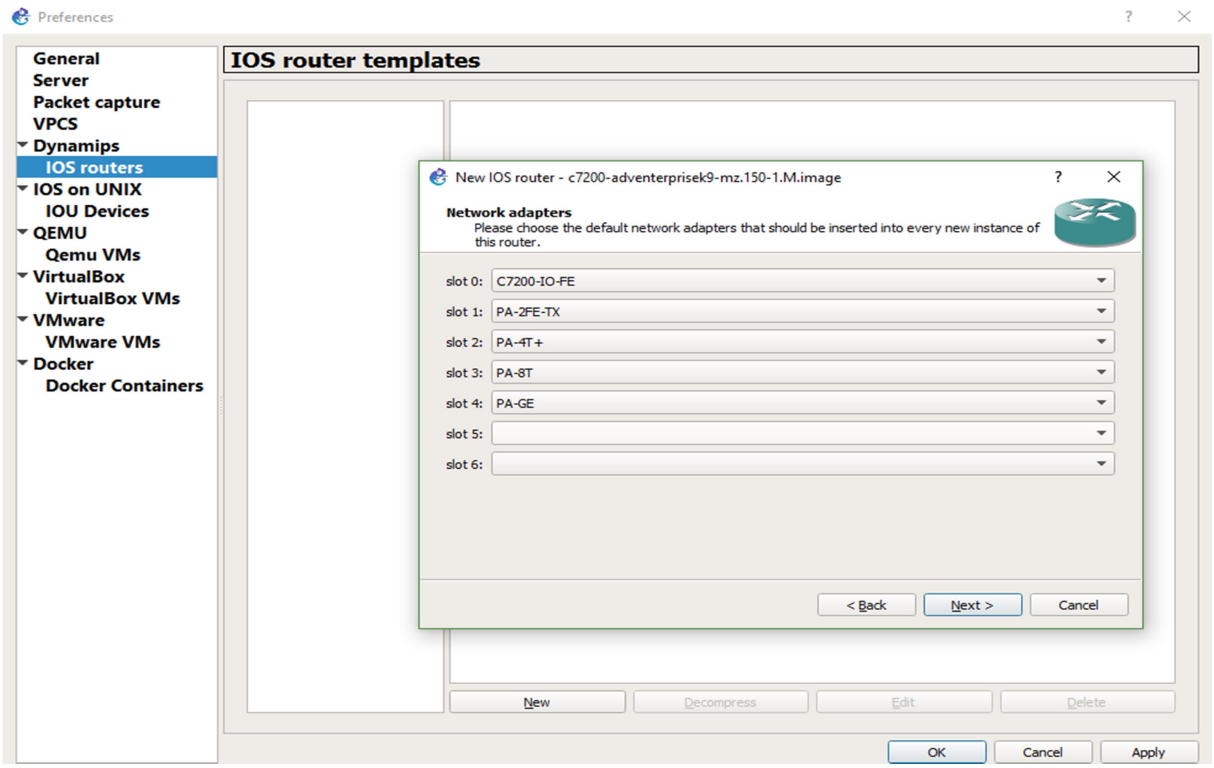


Figura 50. Configuración de Interfaces del router.

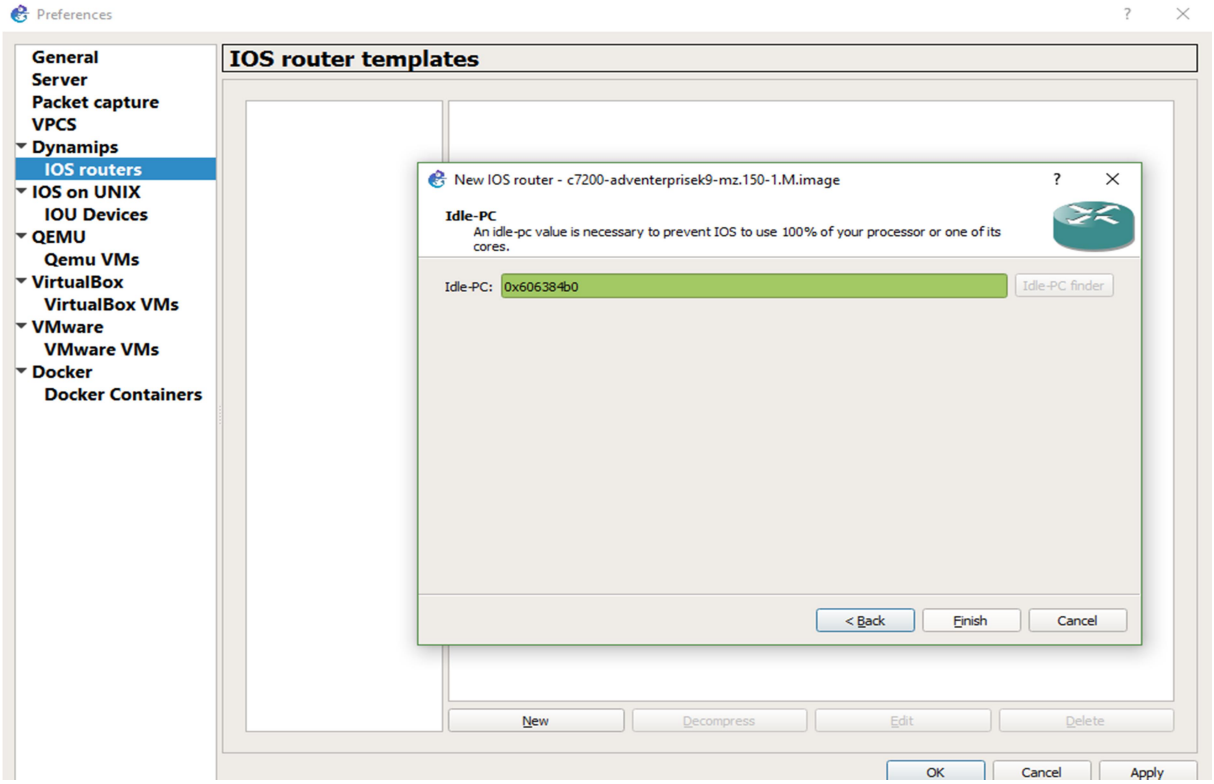


Figura 51. Idle-PC. Depurando la imagen del IOS

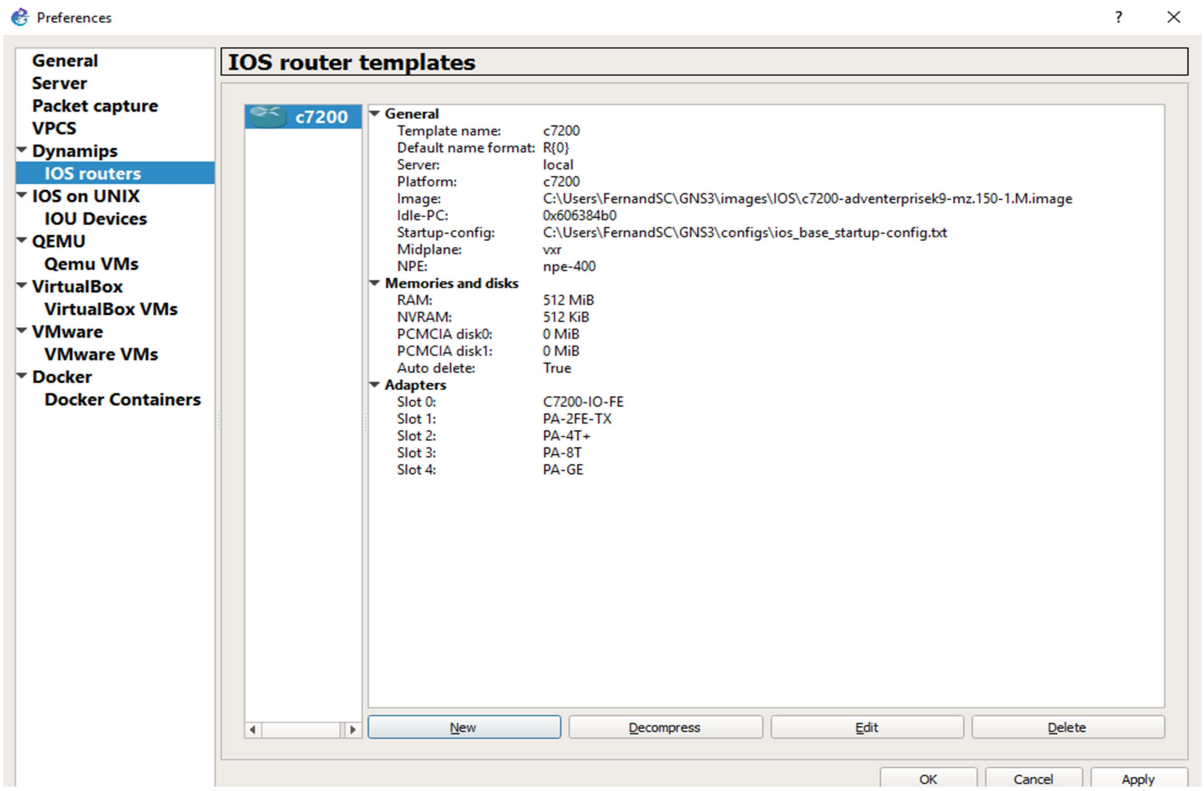


Figura 52. IOS cargado en GNS3

4.3.1. Emulación de la Red GEANT

Una vez configurado el IOS de nuestros routers dispondremos a la conexión de la topología de GEANT. El número de interfaces Giga Ethernet que nos proporciona GNS3 en cada router, es menor respecto a las que nos proporcionó Packet tracer, por ello nos obliga a cambiar el orden de las interfaces en cada router de la red y agregar un router al Pop de “Austria” para cubrir las conexiones de red respecto a la topología de GEANT como se muestra en la figura 53.

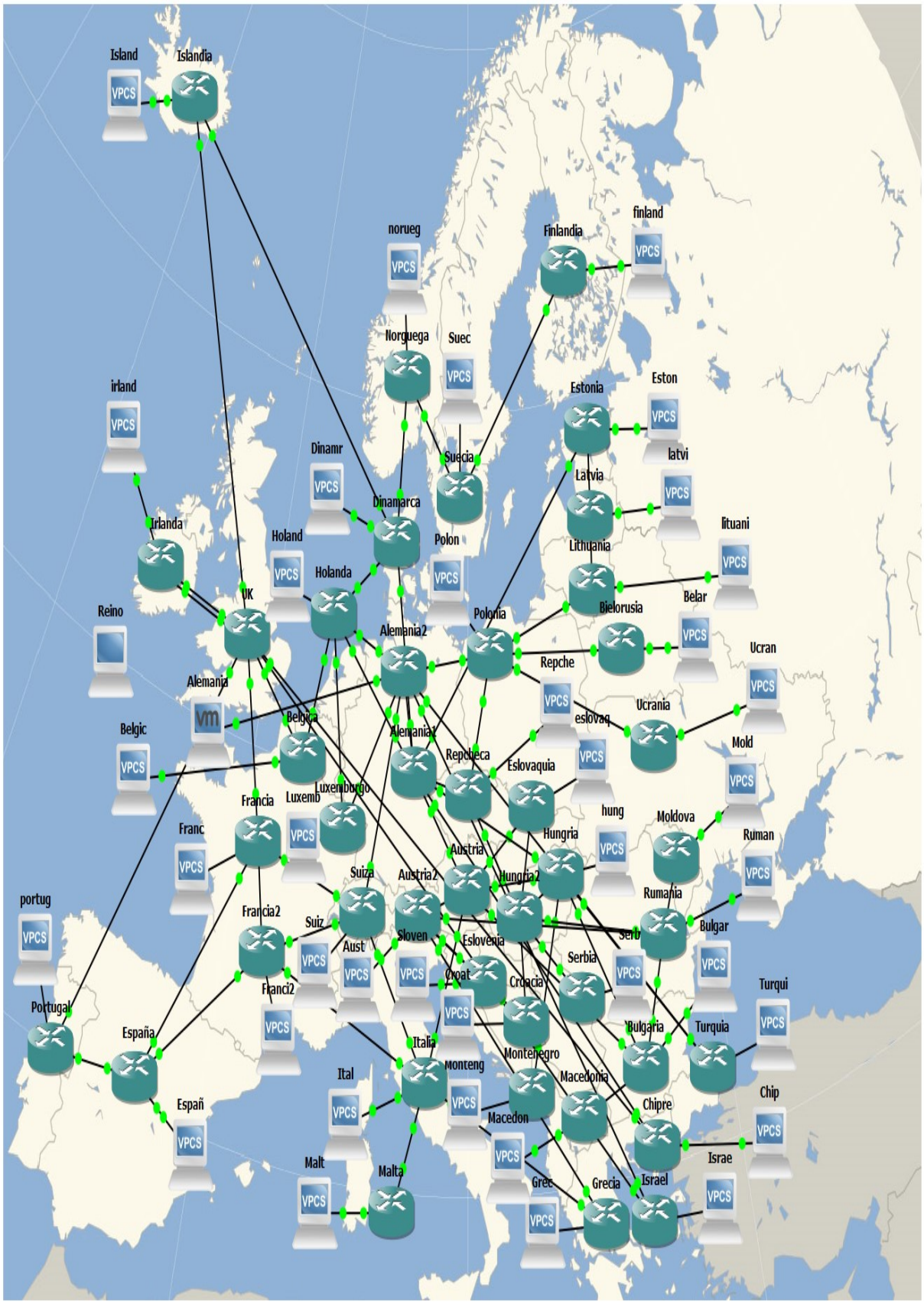


Figura 53. Topología de la red GEANT en emulación en GNS3.

4.3.2. Redes para interfaces de los routers de GEANT para emulación

Como se muestra en la figura anterior se configura un router y un VPCS por cada país europeo. El VPCS (*Virtual Pc Simualtion*) es una simulación de una computadora con código Linux y se configura como host para poder comprobar la conexión entre cada router de la red. Con base en el diseño de redes que propusimos en la simulación, propondremos las conexiones de red de la siguiente manera. Ver tablas 21-28.

Conexión	Red	Interface
Alemania	129.10.2.0	G1/0, G1/0
Holanda	130.10.2.0	G2/0, G2/0
Polonia	131.10.2.0	G3/0, G3/0
Luxemburgo	132.10.2.0	F0/1, F0/1
Suiza	133.10.2.0	G4/0, G4/0
Israel	134.10.2.0	G5/0, G5/0
Turquía	135.10.2.0	G6/0, G6/0

Tabla 21. PoP Alemania 2

Conexión	Red	Interface
Austria	136.10.2.0	G2/0, G2/0
Hungría	137.10.2.0	G3/0, G3/0
Rep. Checa	138.10.2.0	G4/0, G4/0
Chipre	139.10.2.0	G5/0, G5/0
Holanda	140.10.2.0	F0/0, F0/0
Estonia	141.10.2.0	F0/1, F0/1
Dinamarca	142.10.2.0	G6/0, G6/0

Tabla 22. PoP Alemania

Conexión	Red	Interface
Alemania	137.10.2.0	G3/0, G3/0
Hungría 2	159.10.2.0	G1/0, G1/0
Bulgaria	157.10.2.0	G2/0, G2/0
Turquía	158.10.2.0	G4/0, G4/0
Austria	150.10.2.0	G5/0,G5/0
Montenegro	151.10.2.0	G6/0, G6/0

Tabla 23. PoP Hungría

Conexión	Red	Interface
Croacia	152.10.2.0	G2/0, G2/0
Eslovaquia	153.10.2.0	G3/0, G3/0
Rep. Checa	154.10.2.0	G5/0, G5/0
Serbia	155.10.2.0	G6/0, G6/0
Rumania	156.10.2.0	G4/0, G4/0
Hungría	159.10.2.0	G1/0, G1/0

Tabla 24. PoP Hungría

Conexión	Red	Interface
Italia	10.10.2.0	G3/0, G3/0
Alemania	136.10.2.0	G2/0, G2/0
Eslovaquia	12.10.2.0	G4/0, G4/0
Hungría	150.10.2.0	G5/0, G5/0
Bulgaria	14.10.2.0	G6/0, G6/0

Tabla 25. PoP Austria

Conexión	Red	Interface
Croacia	15.10.2.0	G3/0, G3/0
Grecia	16.10.2.0	G2/0, G2/0
Rumania	17.10.2.0	G5/0, G5/0
Eslovenia	18.10.2.0	G4/0, G4/0

Tabla 26. PoP Austria 2.

Conexión	Red	Interface
Irlanda	192.10.2.0	F0/0, F0/0
Irlanda	193.10.2.0	G1/0, G1/0
Francia	194.10.2.0	G2/0, G2/0
Portugal	195.10.2.0	G3/0, G3/0
Bélgica	196.10.2.0	G5/0, G5/0
Chipre	197.10.2.0	G6/0, G6/0
Israel	198.10.2.0	G4/0, G4/0
Islandia	199.10.2.0	F0/1, F0/1

Tabla 27. PoP Reino Unido

Conexión	Red	Interface
Polonia- Lituania	90.10.2.0	G1/0, G1/0
Polonia- Bielorrusia	91.10.2.0	G2/0, G2/0
Polonia- Ucrania	92.10.2.0	G4/0, G4/0
Lituania- Letonia	93.10.2.0	G2/0, G2/0
Letonia – Estonia	94.10.2.0	G3/0, G3/0
Dinamarca- Noruega	95.10.2.0	G3/0, G3/0
Noruega- Suecia	96.10.2.0	G2/0, G2/0
Dinamarca- Holanda	98.10.2.0	G4/0, G4/0
Bélgica- Holanda	99.10.2.0	G6/0, G6/0
Holanda- Luxemburgo	100.10.2.0	G1/0, G1/0
Finlandia- Suecia	101.10.2.0	G3/0, G3/0
Islandia- Dinamarca	22.10.2.0	G2/0, G2/0
España- Portugal	40.10.2.0	G2/0, G2/0
Francia2- España	41.10.2.0	G3/0, G3/0
Francia2 - Francia	42.10.2.0	G1/0, G1/0
España – Francia	43.10.2.0	G4/0, G4/0
Francia- Suiza	44.10.2.0	G3/0, G3/0
Francia2- Suiza	45.10.2.0	G2/0, G2/0
Francia 2- Italia	46.10.2.0	G4/0, G4/0
Suiza- Italia	47.10.2.0	G1/0, G1/0
Malta- Italia	48.10.2.0	G2/0, G2/0
Italia- Grecia	49.10.2.0	G5/0, G5/0
Eslovenia- Grecia	50.10.2.0	G3/0, G3/0
Rep. Checa- Polonia	51.10.2.0	G6/0, G6/0
Macedonia- Bulgaria	52.10.2.0	G3/0, G3/0
Bulgaria-Rumania	53.10.2.0	G1/0, G1/0
Rumania- Moldava	54.10.2.0	G3/0, G3/0

Tabla 28. Resto de conexiones de la red GEANT.

4.3.3. Configuración de Interfaces de los routers de la red GEANT

El procedimiento de la configuración para activar todas las interfaces es el siguiente:

1.- Encender el router.

2.- Entrar al modo configuración global con el comando "*enable*"

3.- Entrar al modo súper usuario con el comando "*conf t*"

4.- Declarar la interface a configurar con el comando "*int*" seguido de la interface a configurar (*interface Giga Ethernet (G0/0) o Fast Ethernet (fa0/0)*).

5.- Declarar la dirección de red y su máscara de red con el comando "*ip add*"

6.- Declarar el ancho de banda para la interface con el comando "*bandwidth*" seguido de valor del ancho de banda

7.- Activar la interface con el comando "no shutdown (no sh)"

8.- Salir del modo súper usuario con el comando "exit"

9.- Guardar los cambios con el comando "*copy run start*" como se muestra en la figura 54.

```
Alemania#en
Alemania#conf t
Enter configuration commands, one per line. End with CNTL/Z.
Alemania (config)# g2/0
Alemania (config-if)# ip add 136.10.2.3 255.255.0.0
Alemania (config-if)# bandwidth 1000000
Alemania (config-if)# no sh
Alemania (config-if)# exit
```

Figura 54. Configuración de interface de red, router Alemania.

Para cercioramos de que estén activas y configuradas las interfaces de manera correcta utilizaremos el comando "*show ip interface brief*", como se muestra en la figura 55.

```
ALEMANIA1#
ALEMANIA1#show ip int brief
Interface                  IP-Address      OK? Method Status
FastEthernet0/0            192.168.4.1     YES NVRAM  up
FastEthernet1/0            unassigned      YES NVRAM  administratively down
FastEthernet2/0            unassigned      YES NVRAM  administratively down
FastEthernet2/1            unassigned      YES NVRAM  administratively down
Serial3/0                  129.10.2.4      YES NVRAM  up
Serial3/1                  136.10.2.3      YES NVRAM  up
Serial3/2                  137.10.2.3      YES NVRAM  up
Serial3/3                  138.10.2.3      YES NVRAM  up
Serial3/4                  139.10.2.3      YES NVRAM  up
Serial3/5                  140.10.2.3      YES NVRAM  up
Serial3/6                  141.10.2.3      YES NVRAM  up
```

Figura 55. Interfaces del router ALEMANIA1

El comando nos despliega la información del tipo de interface, la dirección ip de red configurada y el estatus de cada interface si está habilitada (Up) o deshabilitada (Down).

4.3.4. Configuración de host

Con el objetivo de comprobar comunicación entre los routers, utilizaremos las VPCS que proporciona el emulador como Host para cada router. Para configurar una VPCS ingresamos a modo consola como se muestra en la figura 56.

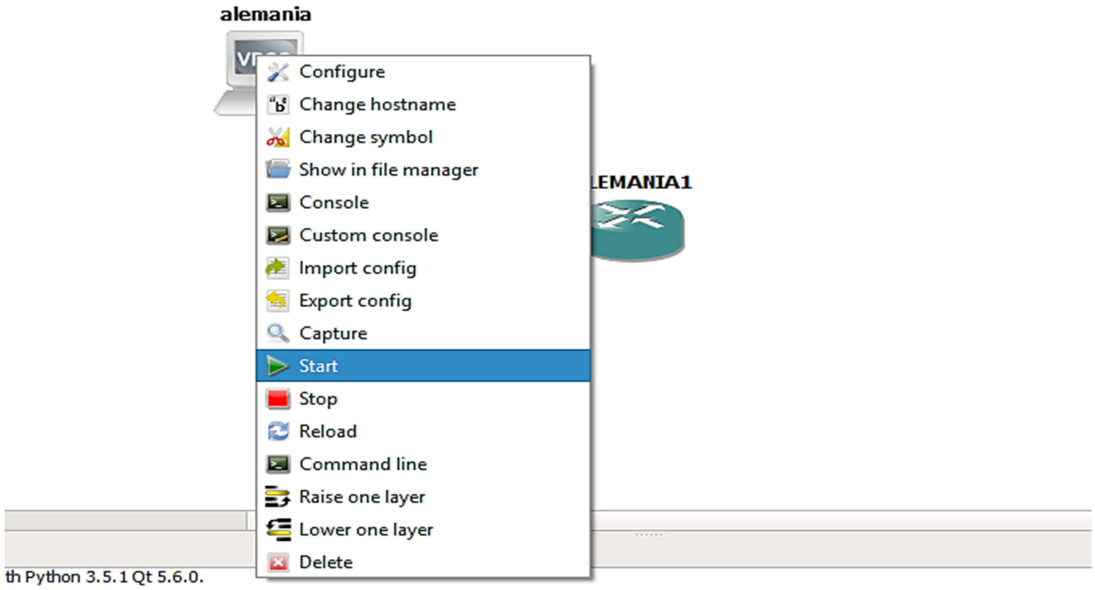


Figura 56. Encendido de VPCS.

Para la configuración de red ingresamos el comando "*ip*" seguido de la dirección IP a configurar, su máscara de red y puerta de enlace, por ultimo ingresamos el comando "*save*" para guardar cambios en el equipo como se muestra en la figura 57.

```
VPCS> ip 192.168.4.2 192.168.4.1 24
Checking for duplicate address...
PC1   : 192.168.4.2 255.255.255.0 gateway 192.168.4.1

VPCS> save
Saving startup configuration to startup.vpc
.  done
```

Figura 57. Configuración de red en VPCS

4.3.5. Configuración del protocolo OSPF de la red GEANT en la emulación

En comparación con el software de simulación, GNS3 solo nos permite la configuración del router vía CLI, es decir que la configuración del protocolo OSPF v2 la realizamos vía comando en cada uno de

los router, dividiendo nuestra red en área 0, área 1 y área 2. Para la configuración del protocolo de enrutamiento seguimos los siguientes pasos:

1.- Encender el router ALEMANIA y entrar en modo comando.

2.- acceder en modo de configuración global "*enable*"

3.- acceder en modo Súper usuario "*conf t*"

4.- Declarar el protocolo OSPF y el proceso: "*route ospf* [*process ID*]"

5.- Declarar el router ID con el comando "router-id [*loopack*]"

6.- Declarar todas las redes que se conectan al router con el comando "*net*" seguido de la dirección de red la wilcard y el área de la red. Como ejemplo mostramos la configuración del router de ALEMANIA1. Ver figura 58.

```
ALEMANIA1#
ALEMANIA1#en
ALEMANIA1#conf t
Enter configuration commands, one per line. End with CNTL/Z.
ALEMANIA1 (config)# router ospf 1
ALEMANIA1 (config-router)# router-id 11.11.11.11
ALEMANIA1 (config-router)# net 192.168.4.0 0.0.0.255 area 0
ALEMANIA1 (config-router)# net 129.10.2.4 0.0.255.255 area 0
ALEMANIA1 (config-router)#
Aug 9 22:16:50.795: %OSPF-5-ADJCHG: Process 1, Nbr 12.12.12.12 on Serial3/0 from
LOADING to FULL, Loading Done
ALEMANIA1 (config-router)# net 139.10.2.4 0.0.255.255 area 0
ALEMANIA1 (config-router)# net 140.10.2.4 0.0.255.255 area 0
ALEMANIA1 (config-router)# net 141.10.2.4 0.0.255.255 area 0
ALEMANIA1 (config-router)# net 142.10.2.4 0.0.255.255 area 0
ALEMANIA1 (config-router)# net 136.10.2.4 0.0.255.255 area 2
ALEMANIA1 (config-router)# net 137.10.2.4 0.0.255.255 area 2
ALEMANIA1 (config-router)# net 138.10.2.4 0.0.255.255 area 2
ALEMANIA1 (config-router)# exit
ALEMANIA1 (config)# exit
ALEMANIA1#
```

Figura 58. Configuración de OSPF en router de Alemania 1.

Cuando configuramos los demás routers de otra área, el protocolo de enrutamiento nos indica que se ha hecho una adyacencia con un router de otra área indicando el número del proceso y el router-id con el que hizo la adyacencia como se muestra en la figura 59.

```
Repcheca#en
Repcheca#conf t

Enter configuration commands, one per line, End with CNTL/Z
Repcheca (Config)#router ospf 1
Repcheca (config-router)#net 192.168.12.0 0.0.0.255 area 2
Repcheca (config-router)#net 138.10.2.0 0.0.255.255 area 2
Repcheca (config-router)#
*Oct 27 10:57:25.839: %OSPF-5-ADJCHG: Process 1, Nbr 7.7.7.7 on GigabitEthernet4/0 from LOADING to FULL, Loadi
ng Done
```

Figura 59. Adyacencia entre routers de Alemania área 0 y Rep. Checa Área 2.

En este ejemplo se configuró el router de Alemania1 y Hungria, los cuales se configuran de manera correcta para tener conectividad entre ambas áreas. Por lo tanto se configuran de la misma manera el resto de routers de la emulación de GEANT.

4.3.6. Configuración de máquinas virtuales GNS3 y VM Ware

El objetivo de crear maquinar virtuales, es diseñar un ambiente lo más cercano a la realidad, probar conectividad y configurarla como una estación de gestión SNMP utilizando aplicaciones como *PowerSNMP Free Manager* y MIB Browser Personal.

Para crear una máquina virtual necesitamos el software VMware Workstation, previamente instalado y seguir los pasos siguientes:

1.- Abrir el programa VMware.

2.- Abrir la opción de la pestaña *file* → *New Virtual Machine*→ *Typical*→

3.- Seleccionar la ruta de la imagen (ISO) del sistema operativo.

4.- Elegir el tipo de sistema operativo, en este caso instalaremos Windows 7 de Microsoft → Nombre de nuestra máquina virtual (Alemania)

5.- Elegir el volumen de nuestro disco duro virtual para nuestra máquina. Recomendamos que sea superior a 7 Gb dado que el software de Gestión SNMP necesita distintos componentes para su funcionamiento.

6.-Elegimos la opción "*Split virtual into multiple files"*, para que nuestro disco virtual se divida en múltiples archivos.

7.- Creamos nuestra máquina virtual con la opción *Finish* como se muestra en la figura 60.

Figura 60. Creación de máquina virtual Alemania.

Una vez instalado nuestro sistema operativo configuraremos la NIC de la máquina virtual, de la MV de Alemania, la encendemos→ inicio → panel de control → Redes e Internet → Centros de redes y recursos Compartidos→ cambiar configuración del adaptador → click derecho en conexión de red de área local → propiedades→ protocolo de internet versión 4 (IPV4) → usar la siguiente dirección IP y llenar los campos con los datos de la tabla 26 como se muestra en la figura 61 y 62.

Dirección IP	192.168.4.2
Mascara de Subred	255.255.255.0
Puerta de enlace Predeterminada	192.168.4.1

Tabla 61. Parámetros para la configuración de red.

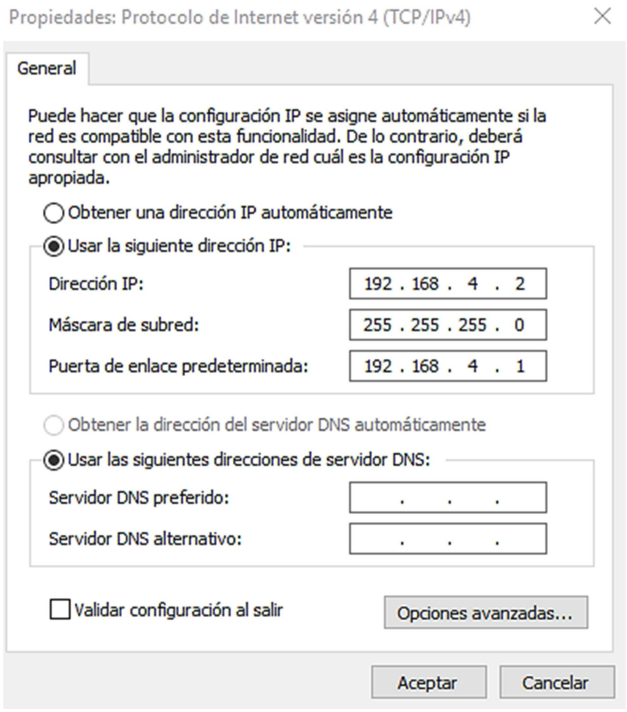


Figura 62. Configuración de la NIC de la MV de Alemania.

Para evitar un problema de conectividad en nuestra emulación de red, desactivamos el *Firewall* de nuestra máquina virtual.

4.3.7. Configuración de máquinas virtuales con GNS3

Una vez creada la máquina virtual procedemos a su configuración para vincularlas con GNS3. El primer paso es abrir GNS3 seleccionamos → *Edit* → *Preferences* → *VMware* → *VMware VMs*. Ver figura 63.

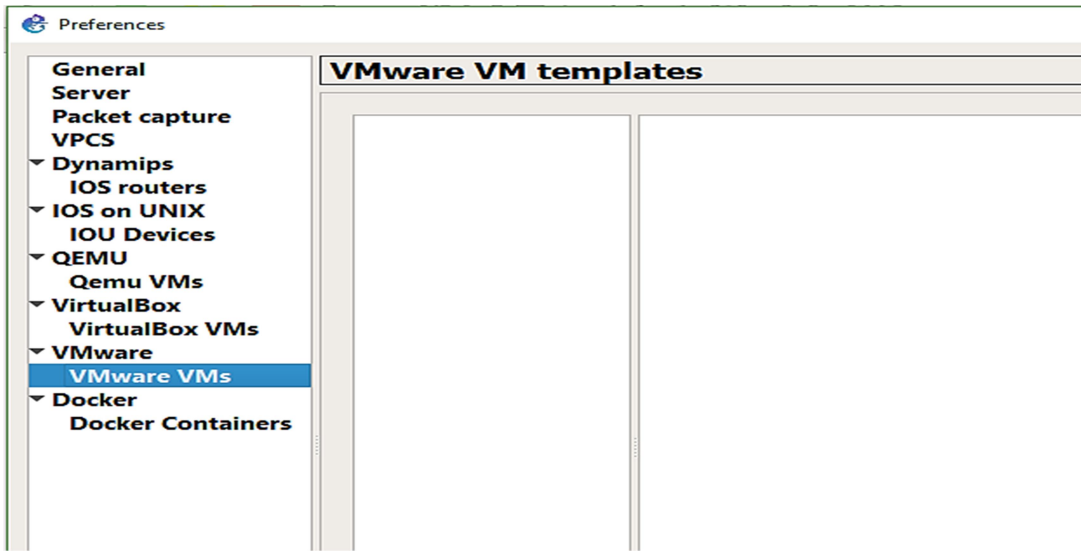


Figura 63. Configuración de VMware en GNS3.

Seleccionamos "*New*" y cargamos nuestra máquina virtual que generamos anteriormente en VM Ware y damos click en "*Finish*". Ver figura 64 y 65.

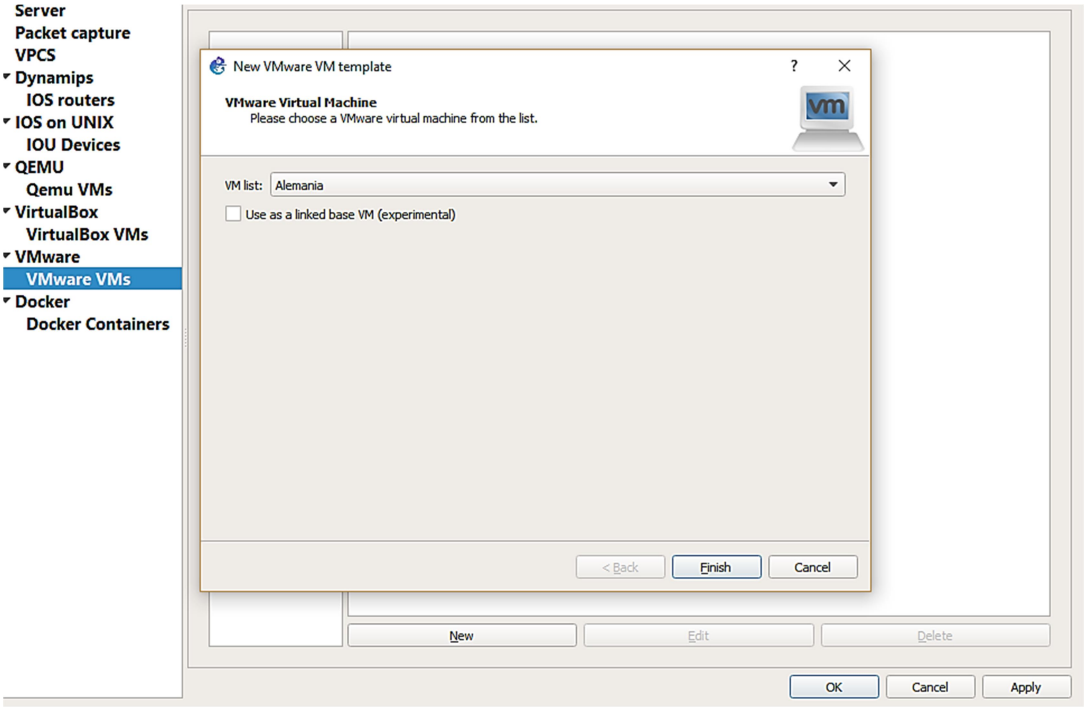


Figura 64. Vinculación de VM de Alemania en GNS3.

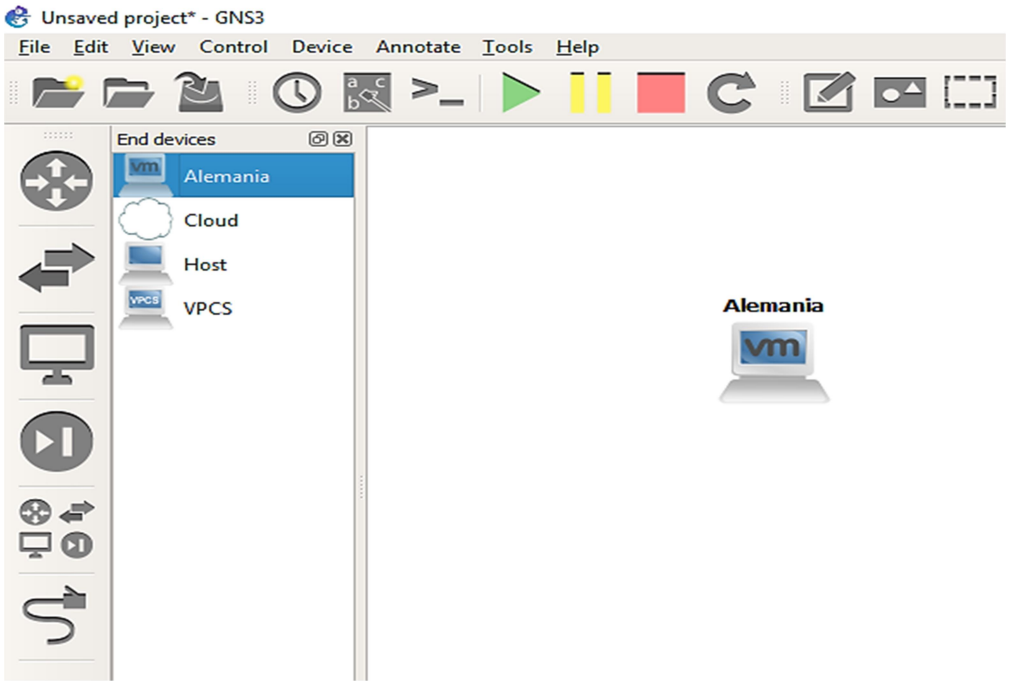


Figura 65. Máquina Virtual vinculada en GNS3.

4.3.8. Configuración de redes Virtuales VMware y GNS3

Una vez que agregamos nuestra máquina virtual a GNS3 necesitamos crear los vínculos de red, es decir, crear redes virtuales para que nuestro emulador y la máquina virtual se conecten. Para esto configuramos GNS3, seleccionamos *Edit → Preferences → VMware → network*. Ver figura 66.

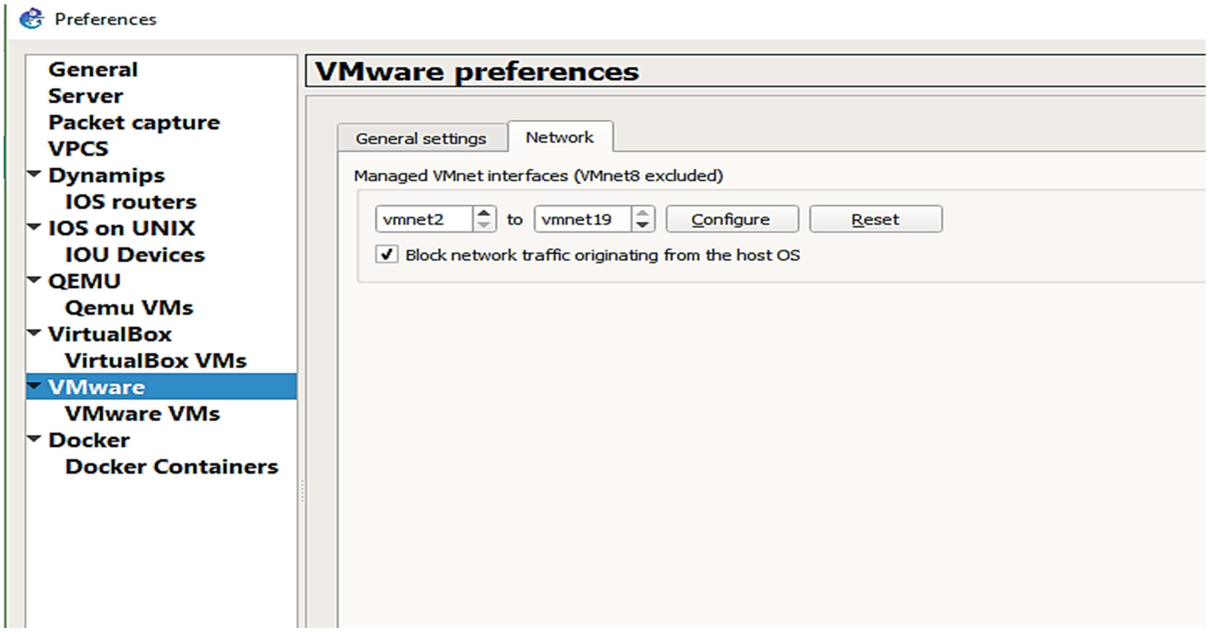


Figura 66. Creación de Vmnet-Interfaces de red virtuales.

En este paso vamos a crear interfaces virtuales de red para que pueda vincularse GNS3 con VMware. Se pueden configurar de 1 hasta 19 interfaces, dependiendo las necesidades de nuestra emulación. Para crearlas elegimos el número de VMnet y damos click en "*configure*" ver figura 67.

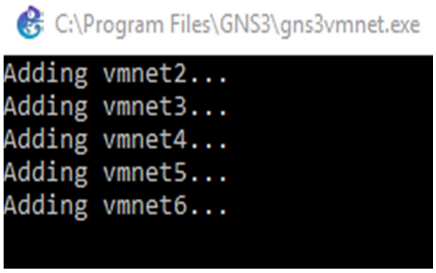


Figura 67. Agregando Vmnet.

Una vez agregadas nuestras *VMnet* procederemos a cambiar la interfaz de red en nuestra máquina virtual, es decir configurar la *VMnet* que elegimos en nuestro *Virtual Network Editor (VMnet2).* Abrimos VMware, en la barra de *Lybrary,* elegimos la máquina virtual que creamos (Alemania) y elegimos la opción *New Adapter*, cambiamos la opción a *Custom Specific Virtual Network* y elegimos la VMnet2. Ver figura 68.

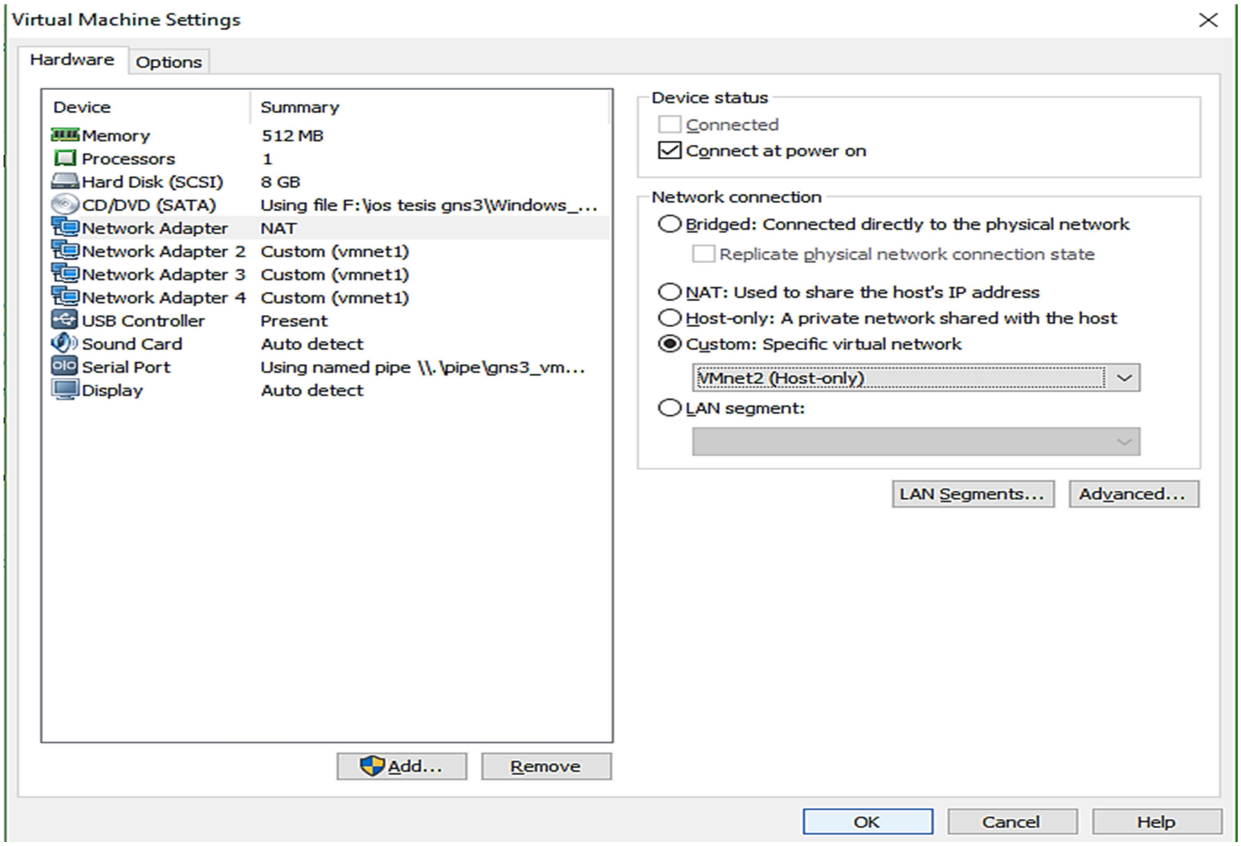


Figura 68. Configuración de redes virtuales.

4.3.9. Configuración de SNMP en la emulación de la red GEANT

Una vez configurado el enrutamiento en nuestra emulación, el siguiente paso consiste en configurar SNMP y definir nuestro sistema de gestión de red, por lo que elegimos al host de Alemania 2 y los dispositivos gestionados. En este caso configuraremos como ejemplo el router de Alemania 2 como se muestra en la figura 69.

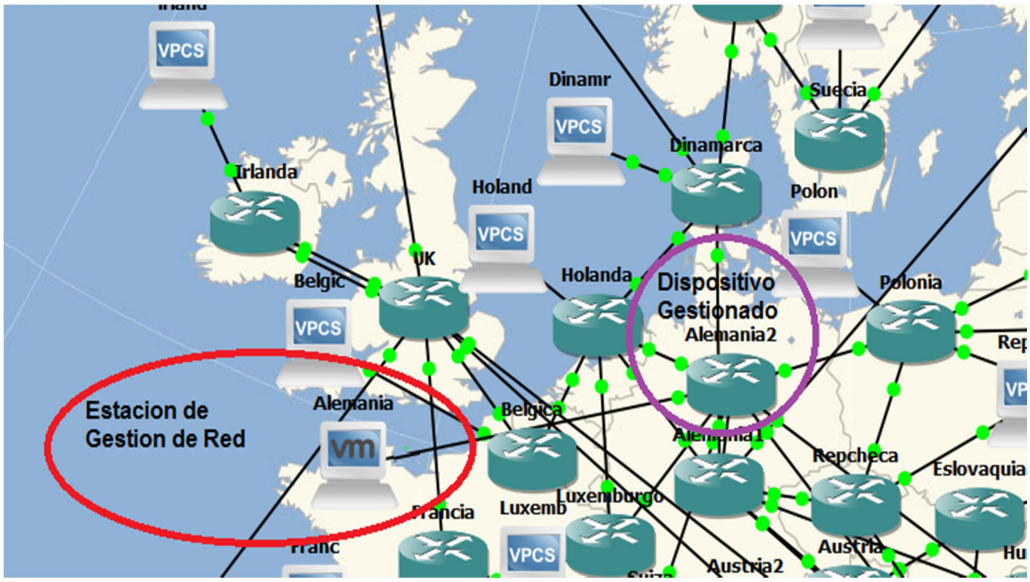


Figura 69. Elementos de Gestión SNMP en emulación.

4.4. Configuración de la estación de gestión de red en la emulación de GEANT

El proceso de Gestión en la Emulación de la red GEANT será distinto dado que el software de emulación no cuenta con una interfaz gráfica de gestión como el MIB Browser de Packet Tracer, entonces será necesario instalar un software de gestión SNMP en nuestra máquina virtual que permita la gestión y monitoreo de los agentes.

4.4.1. Power SNMP *Free Manager* y GNS3

El software *Power SNMP free Manager* está diseñado para consultar y supervisar los valores de las variables de los agentes SNMP, recibir traps de los agentes y configurar las alertas con notificaciones por correo electrónico. Para ello instalaremos *PowerSNMP Free Manager*, en la máquina virtual de Alemania como se muestra en la figura 70.

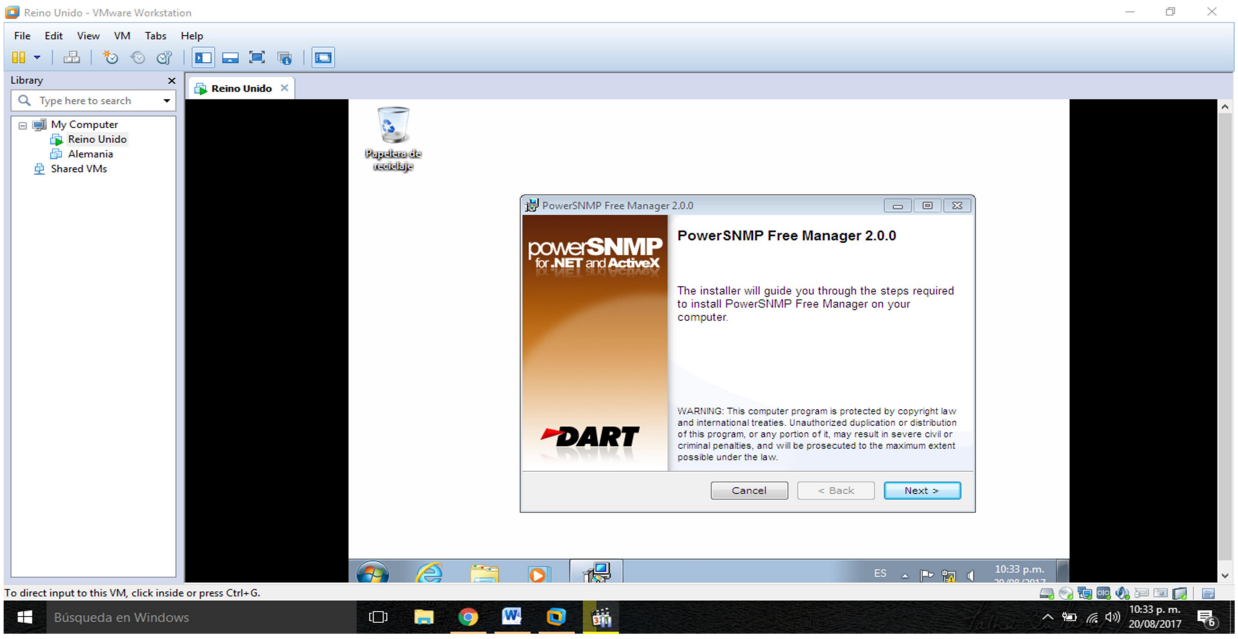


Figura 70. Instalación de Power SNMP Free Manager

4.4.2. Configuración de SNMP en GNS3

Una vez instalada nuestra aplicación de gestión en la máquina virtual procederemos a configurar SNMP en el router de Alemania 2 por medio de líneas de comando, declaramos la comunidad (*geant*), que usamos en la simulación como se muestra en la figura 71.

```
Alemania2#
Alemania2#en
Alemania2#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Alemania2(config)#snmp-server community geant ro SNMP_ACL
Alemania2(config)#snmp-server community geant rw SNMP_ACL
Alemania2(config)#snmp-server host 192.168.4.2 version 2c geant
Alemania2(config)#snmp-server enable traps
% Cannot enable both sham-link state-change interface traps.
% New sham link interface trap not enabled.
Alemania2(config)#ip access-list standard SNMP_ACL
Alemania2(config-std-nacl)#permit 192.168.4.2
Alemania2(config-std-nacl)#exit
```

Figura 71. Configuración de SNMP v2 en router Alemania 2.

Una vez configurado el protocolo SNMP en el agente del router, configuraremos *Power SNMP Free Manager* para agregar los routers gestionados, seleccionamos la opción de configuración, agregamos la *Gateway* de la VM de Alemania 2 en el campo *"Local Address "y* configuramos el puerto 162 para el envio de *Traps* como se muestra en la figura 72.

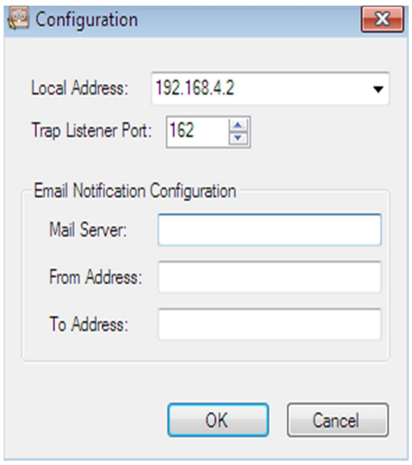


Figura 72. Configuración del puerto.

Para descubrir el agente del dispositivo gestionado por parte de *Power SNMP free Manager* elegimos la opción "*discover*" →*SNMP Agents* → y agregaremos la dirección de red en difusión 192.168.4.255 para encontrar el agente configurado, como se muestra en la figura 73.

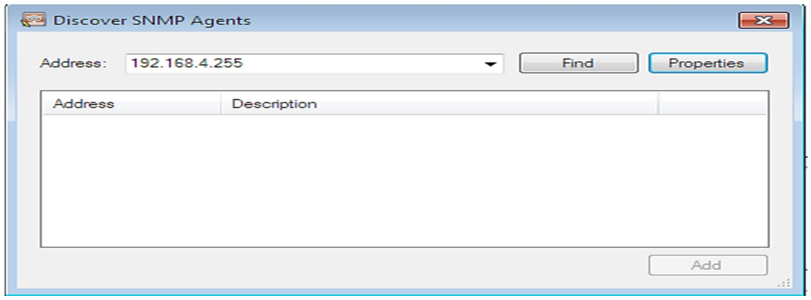


Figura 73. Descubriendo Agentes SNMP.

Posteriormente configuramos la seguridad basada en comunidad seleccionando "*properties*" y en el campo "*community*" agregamos la comunidad configurada en el agente del router y por último seleccionamos la versión 2 del protocolo SNMP. Ver figura 74.

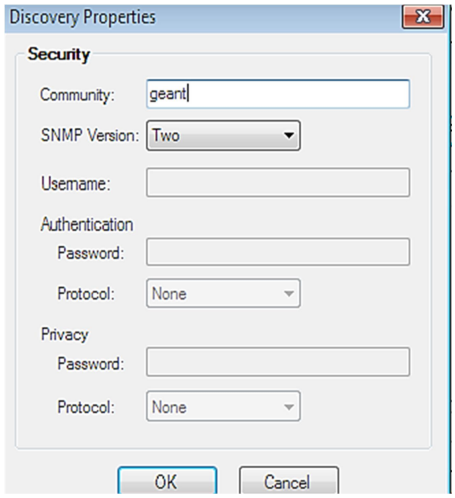


Figura 74. Configuración de Comunidad.

Realizada la configuración de los agentes en *Power SNMP Free manager*, estos son detectados como se muestra en la figura 75.

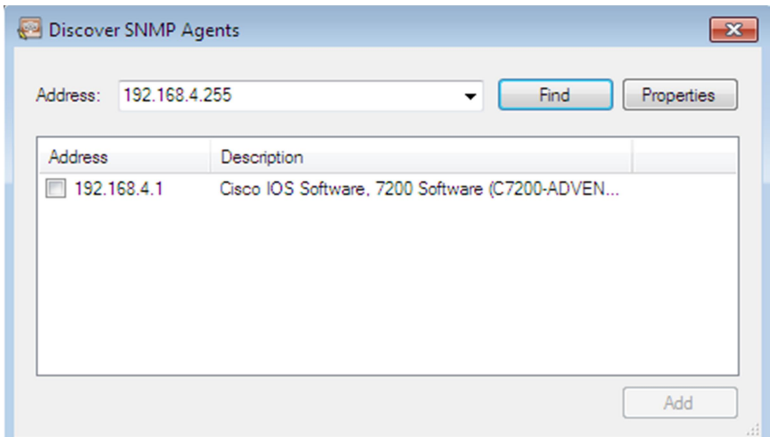


Figura 75. Agente del router de Alemania en SNMP en *Power SNMP Free Manager*.

Una vez detectados los agentes en la aplicación, estos ya están habilitados para enviar y recibir traps, por medio de un cuadro de dialogo de *Traps,* registrando el tipo de llegada de la alerta, el origen y el tipo de OID. También, podemos analizar los objetos del agente por medio del árbol MIB que nos despliega en la parte derecha superior la aplicación SNMP como se muestra en la figura 76.

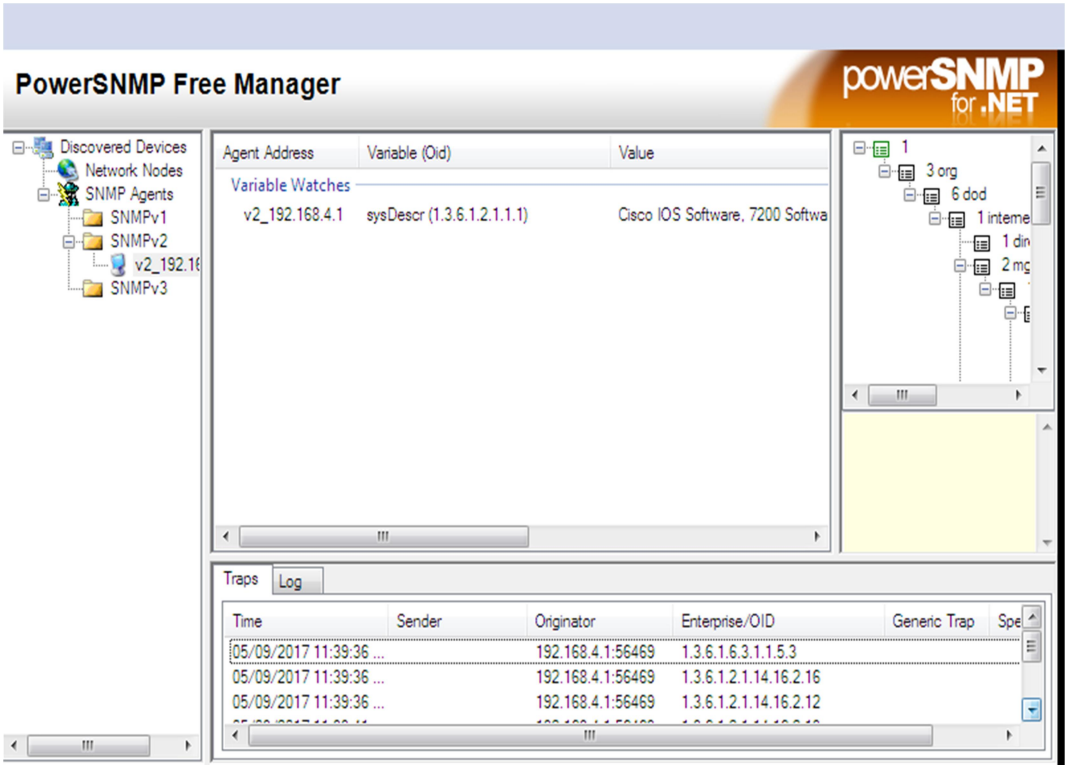


Figura 76. Agente del router Alemania 2 vinculado en *Power SNMP Free Manager*.

4.4.3. Configuración de SNMP v3 en Emulación

Para la configuración del protocolo de gestión SNMP v3 es necesario configurar la entidad en el router. Por lo que accedemos en modo súper usuario y declaramos el comando del protocolo SNMP con el nombre del grupo llamado "*Advnetlab*", usando privacidad SNMP v3como se muestra en la figura 77.

```
R2#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R2(config)#snmp-server group Advnetlab ?
  v1    group using the v1 security model
  v2c   group using the v2c security model
  v3    group using the User Security Model (SNMPv3)

R2(config)#snmp-server group Advnetlab v3 ?
  auth     group using the authNoPriv Security Level
  noauth   group using the noAuthNoPriv Security Level
  priv     group using SNMPv3 authPriv security level

R2(config)#snmp-server group Advnetlab v3 priv ?
  access    specify an access-list associated with this group
  context   specify a context to associate these views for the group
  match     context name match criteria
  notify    specify a notify view for the group
  read      specify a read view for the group
  write     specify a write view for the group
  <cr>

R2(config)#snmp-server group Advnetlab v3 priv
```

Figura 77. Configuración de SNMP v3. Tipos de Seguridad.

Posteriormente declaramos la identidad, es decir, el nombre del usuario SNMP v3, en este caso será "*uacm*", también escogeremos el tipo de autentificación MD5 con la clave "t1tul4c10n" y la privacidad con el cifrado DES con contraseña "dosm1118" como se muestra en la figura 78.

```
R1(config)#snmp-server user uacm Advnetlab v3 auth md5 t1tul4c10n ?
  access  specify an access-list associated with this group
  priv    encryption parameters for the user
  <cr>

R1(config)#snmp-server user uacm Advnetlab v3 auth md5 t1tul4c10n priv ?
  3des  Use 168 bit 3DES algorithm for encryption
  aes   Use AES algorithm for encryption
  des   Use 56 bit DES algorithm for encryption

R1(config)#snmp-server user uacm Advnetlab v3 auth md5 t1tul4c10n priv des ?
  WORD  privacy pasword for user

R1(config)#$ user uacm Advnetlab v3 auth md5 t1tul4c10n priv des dosm1118 ?
  access  specify an access-list associated with this group
  <cr>

R1(config)#$ user uacm Advnetlab v3 auth md5 t1tul4c10n priv des dosm1118
R1(config)#
*Jan  8 16:10:28.703: Configuring snmpv3 USM user, persisting snmpEngineBoots. Please Wait...
```

Figura 78. Configuración de seguridad SNMP v3.

Por ultimo configuramos los mensajes de tipo Get y Set en la version 3 de SNMP usando los comandos que se muestran en la figura 79.

```
Alemania2(config)#snmp-server group Advanetlab v3 priv
Alemania2(config)#snmp-server user uacm Advanetlab v3 auth ?
  md5  Use HMAC MD5 algorithm for authentication
  sha  Use HMAC SHA algorithm for authentication

Alemania2(config)#$ user uacm Advanetlab v3 auth md5 tltul4cl0n priv?
priv

Alemania2(config)#$ user uacm Advanetlab v3 auth md5 tltul4cl0n priv des?
des

Alemania2(config)#$ Advanetlab v3 auth md5 tltul4cl0n priv des dosml118
Alemania2(config)#
*May  7 17:57:28.107: Configuring snmpv3 USM user, persisting snmpEngineBoots. Ple
ase Wait...

Alemania2(config)#snmp-server group Advanetlab v3 auth read?
read

Alemania2(config)#snmp-server group Advanetlab v3 auth read V3Read?
WORD

Alemania2(config)#$ group Advanetlab v3 auth read V3Read write V3Write
Alemania2(config)#snmp-server view V3Read iso included
Alemania2(config)#snmp-server view V3Write iso included
Alemania2(config)#snmp-server host 192.168.4.2 version 3 auth uacm
Alemania2(config)#snmp-server user uacm Advanetlab v3 auth md5 tltul4clclon
Alemania2(config)#exit
Alemania2#exit
```

Figura 79. Configuración de *Get y Set* SNMP v3.

Para la configuración de la entidad en *Power SNMP Free Manager*. Elegimos la version 3 y después llenamos los campos del nombre de usuario, la clave de autentificación y la clave privada, con los datos configurados en la entidad del router de Alemania 2 como se muestra en la figura 80.

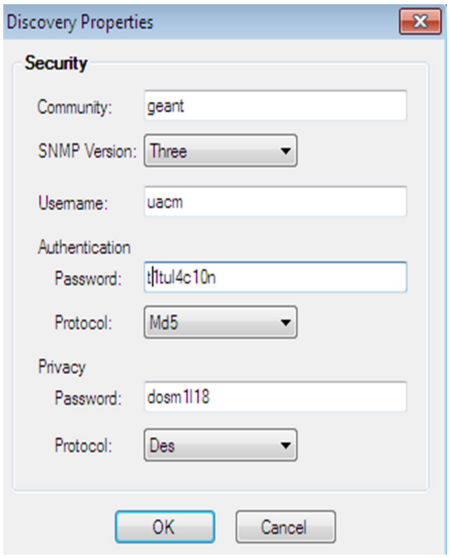


Figura 80. Configuración de la entidad SNMP v3 en *Power SNMP Free M.*

Una vez configurada la entidad SNMP v3 en *Power SNMP Free Manager* comprobamos que ha sido detectada con éxito como se muestra en la figura 81.

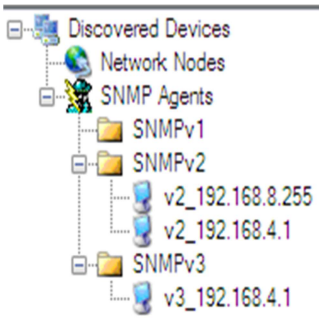


Figura 81. Entidad SNMP v3 en *Power SNMP Free Manager.*

Capítulo 5

Resultados

5.1.1. Prueba de Conexión en la simulación de la Red Avanzada GEANT

Para probar que nuestras configuraciones de enrutamiento fueron correctas, utilizaremos el comando "ping", el cual nos ayudara a comprobar conectividad en la red por medio del protocolo ICMP (*Internet Control Message Protocol*). El simulador Packet Tracer nos permite realizar la prueba PING de manera gráfica y con línea de comando.

5.1.2. Prueba de conectividad en modo CLI

Para realizar la prueba de conectividad con el comando *ping*, en modo comando accedemos a la consola del router o de nuestro host de origen. En este caso enviaremos la prueba del host de Portugal al host de UK, ingresando el comando "ping" seguido de la dirección IP destino como se muestra en la figura 82:

Ejemplo:

Origen: Host de Portugal.
Destino: Host de UK.
Comando:
PORTUGAL> ping 192.168.36.2

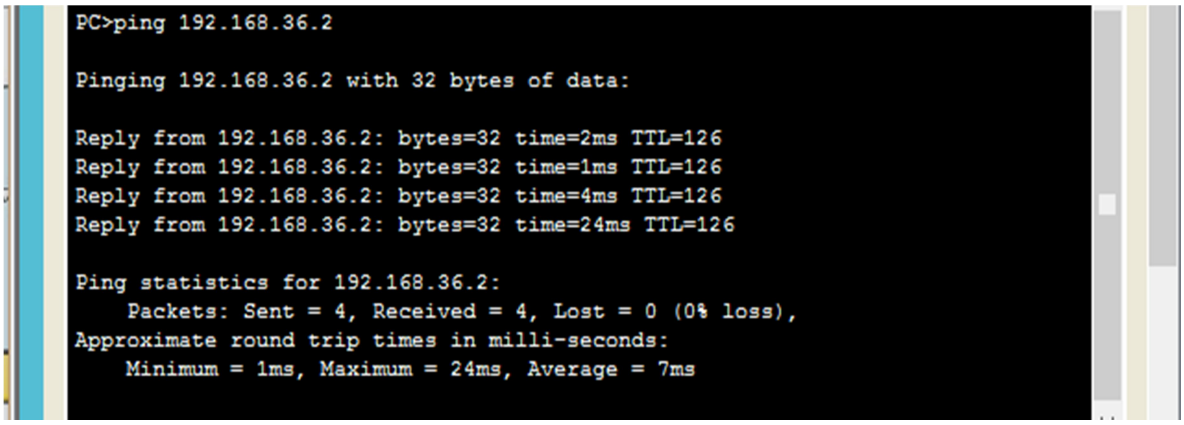


Figura 82. Prueba ping en modo CLI del host de Portugal al host de UK.

La figura 78 nos muestra la consola del host de Portugal realizando la prueba *ping* la cual nos arroja un breve resumen del número de paquetes enviados y perdidos, el tiempo promedio por paquete enviado y el tiempo de vida de cada paquete, Time To Live (TTL). Además nos muestra que el número de

paquetes recibidos son 4 y el número de paquetes perdidos son 0 por lo que nos demuestra que nuestra conectividad fue exitosa.

5.1.3. Prueba de conectividad en modo GUI en Simulación

Para entender mejor el proceso de la prueba PING, el simulador Packet Tracer nos ofrece una interfaz gráfica marcando la ruta que siguen los paquetes. Tomando el ejemplo anterior en modo CLI, realizamos la prueba PING entre el host de Portugal y el host de UK como se muestra en la figura 83.

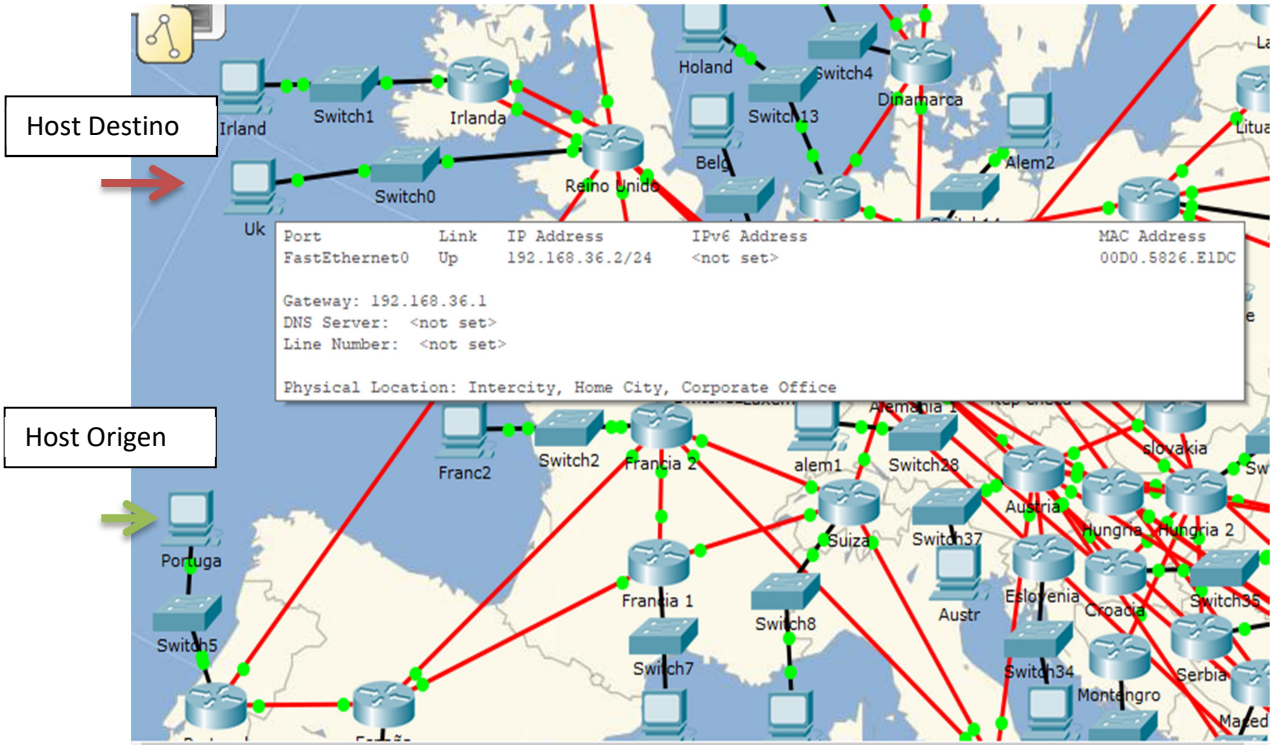


Figura 83. Prueba PING en Modo GUI entre UK y Portugal.

Este proceso puede variar en tiempo dependiendo del protocolo de enrutamiento y el tamaño de la red. Esto se le conoce como latencia, es decir, el retardo o tiempo que transcurre en el envío y transmisión de paquetes en una red.

5.1.4. Mensajes ARP e ICMP en prueba de conectividad

La prueba Ping envía mensajes de tipo ICMP. Sin embargo, cuando existe por vez primera comunicación se envían mensajes de tipo ARP. Los mensajes de tipo ICMP principalmente se usan para detectar y reportar algún error en la red verificando si los equipos conectados en la red están habilitados por medio de mensajes de host a host ICMP petición-respuesta.

Del mismo para que exista comunicación entre estos equipos es necesario tener un registro de las direcciones físicas y lógicas de los equipos en la red, es por ello se envían mensajes de tipo ARP, que se encargan de recolectar información de la dirección física MAC (*Media Access Control*) de los equipos y relacionarlas con sus direcciones lógicas (dirección IP) respectivas para crear la tabla ARP y realizar un mapa dinámico de la red. Para entender mejor este proceso explicaremos con la ayuda del simulador que pasa cuando realizamos la prueba PING entre el host de Portugal y UK.

Al enviar el primer paquete, este envía un mensaje el cual solo se lleva información de la dirección IP Origen como se muestra en la figura 84.

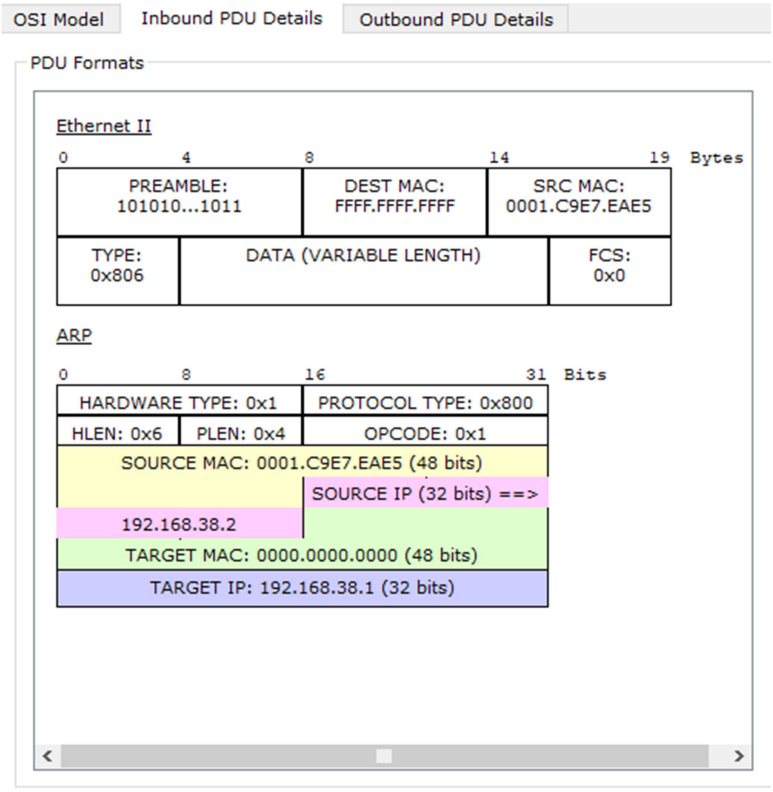


Figura 84. Formato de mensaje, Direcciones fuente, destino.

En este primer mensaje nos damos cuenta que no existe información de las direcciones MAC de los equipos conectados a la red, por lo tanto se enviará un mensaje ARP para hacer petición de las direcciones físicas de los equipos conectados como se muestra en la figura 81.

Una vez creada la adyacencia entre ambas redes el router destino comparte la información de su dirección lógica regresando un mensaje similar con la información solicitada para crear la tabla de direcciones físicas por medio del protocolo ARP como se muestra en la figura 85-86.

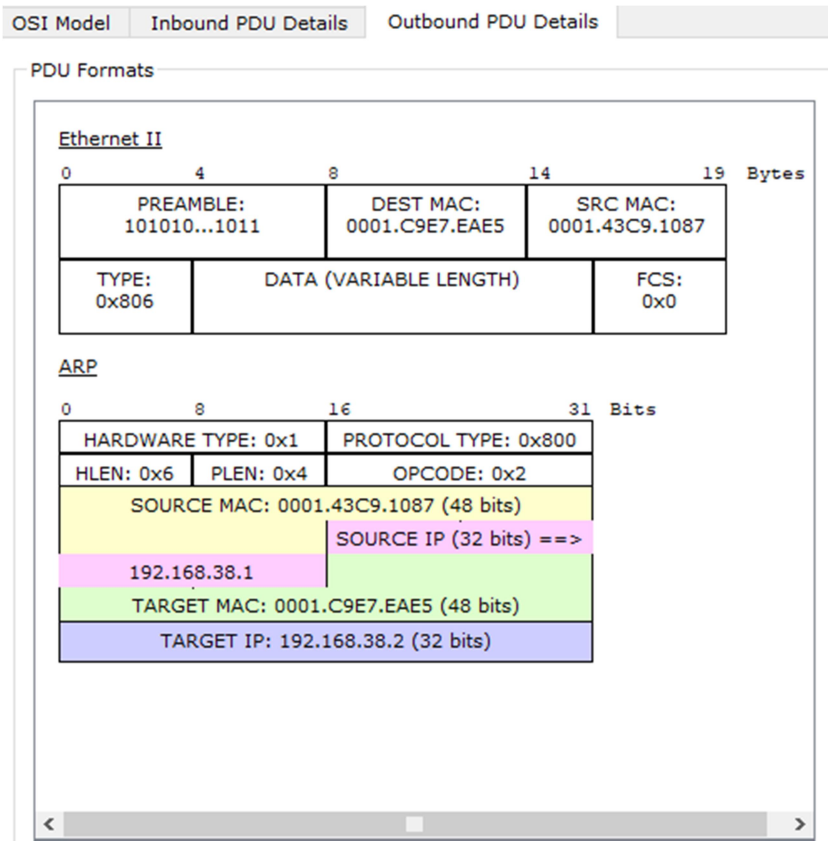


Figura 85. Mensaje respuesta envio de dirección MAC destino.

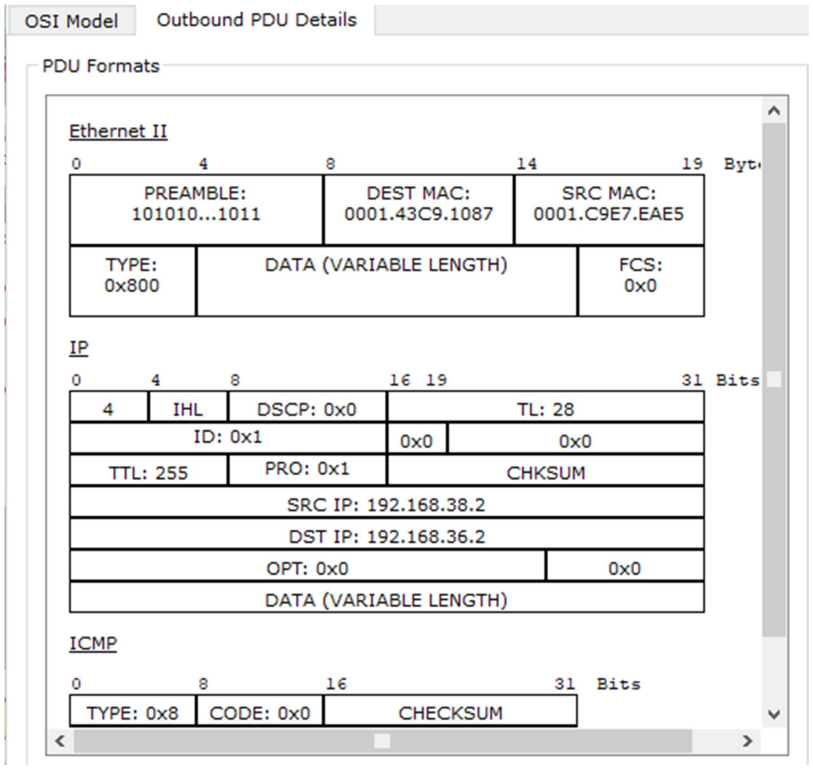


Figura 86. Mensaje IP con direcciones físicas y lógicas Origen- Destino.

El envío de mensajes ARP se repite dependiendo del número de equipos conectados en la red. En este caso se logra obtener con éxito la dirección MAC destino y el simulador crea la tabla ARP.

Después del proceso ARP, el router ya tiene información de la red para crear una ruta y mandar el paquete al equipo de destino. La elección de la ruta destino depende tanto de la tabla de enrutamiento y del protocolo que estemos ocupando.

Por medio de los formatos de mensajes del simulador y con el proceso ISO/OSI, seguimos la ruta de nuestro paquete hasta llegar al host de UK. La figura 83 nos indica que nuestro paquete se envía por la ruta de la red *195.10.2.3*, la cual es la interfaz de red que conecta a Portugal con UK como se muestra en la figura 87 y 88.

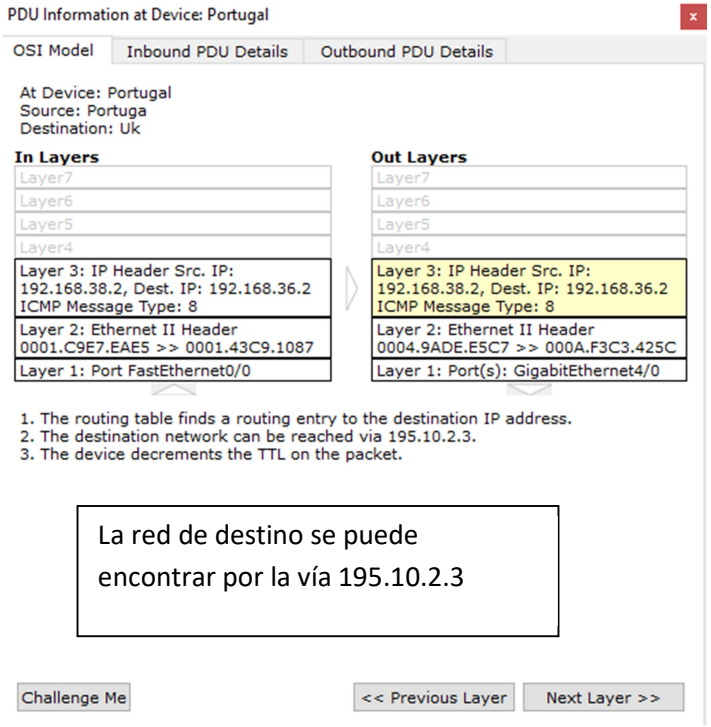


Figura 87. Mensaje PDU Portugal.

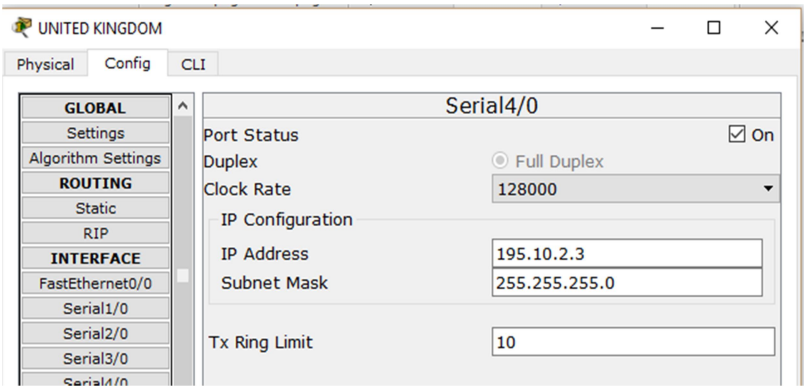


Figura 88. Interfaz de red que conecta a Portugal con UK.

El proceso anterior se repite dado que son mensajes de petición respuesta y también dependen del número de elementos de red que existan. Una vez estructurada la tabla ARP, los mensajes ICMP comprueban que no exista algún error de conectividad en la red, si la conectividad se realizó de manera correcta se recibe un mensaje de tipo respuesta el cual nos indica que se envió el mensaje con éxito al host destino, como se muestra en la figura 89.

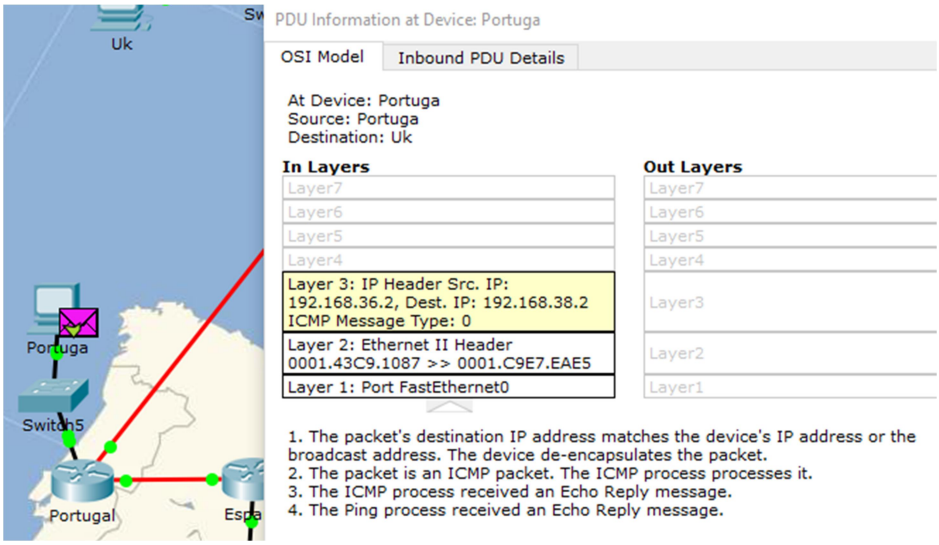


Figura 89. Prueba PING con éxito de UK a Portugal.

5.1.5. Prueba de conectividad en RIP v2 en la simulación de GEANT

Configurado el protocolo RIP v2 probaremos conectividad, pondremos a prueba su métrica, la convergencia y latencia de los paquetes enviados por su red. Además se comparará su funcionamiento con el del protocolo OSPF. Las primeras pruebas de conectividad que se realizaron fueron desde el host de Alemania 2 hacia todos los PoP de la red. Esta prueba se realizó en modo comando como se muestran en las figuras 90-91.

```
PC>ping 192.168.10.2

Pinging 192.168.10.2 with 32 bytes of data:

Request timed out.
Reply from 192.168.10.2: bytes=32 time=0ms TTL=125
Reply from 192.168.10.2: bytes=32 time=190ms TTL=125
Reply from 192.168.10.2: bytes=32 time=16ms TTL=125

Ping statistics for 192.168.10.2:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 190ms, Average = 68ms

PC>ping 192.168.39.2

Pinging 192.168.39.2 with 32 bytes of data:

Request timed out.
Reply from 192.168.39.2: bytes=32 time=0ms TTL=125
Reply from 192.168.39.2: bytes=32 time=0ms TTL=125
Reply from 192.168.39.2: bytes=32 time=12ms TTL=125

Ping statistics for 192.168.39.2:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 12ms, Average = 4ms
```

Figura 90. Prueba de conectividad para Austria y Bélgica.

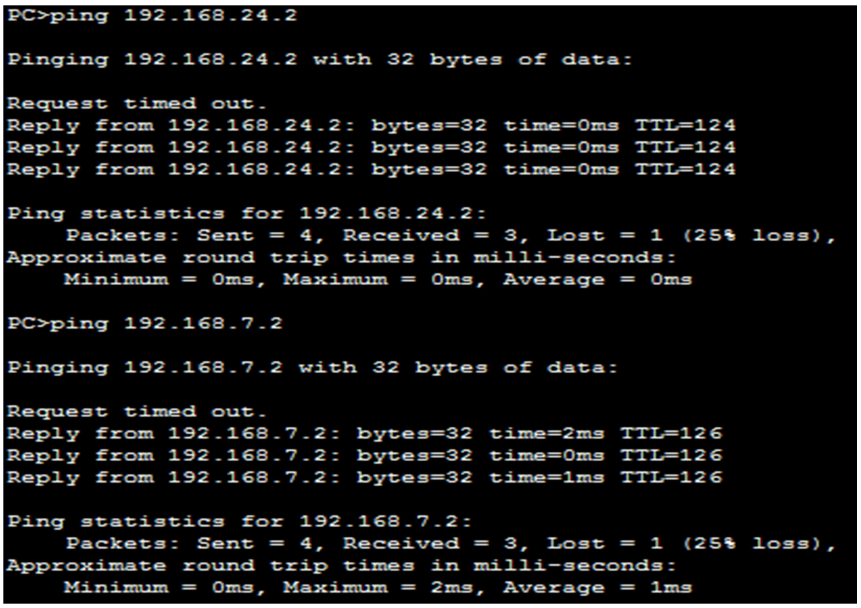


Figura 91. Prueba de conectividad para Bulgaria y Suiza

La conectividad al resto de los nodos de la red, fueron probados con éxito y las evidencias se encuentran en el apéndice A.

5.1.6. Métrica del Protocolo RIP v2 en la simulación de GEANT

Con base en el algoritmo vector distancia y la métrica de conteo de saltos verificamos que el protocolo RIP v2 se pudo aplicar en la simulación de la red GEANT ya que su diseño distribuido de la red, permitió que algoritmo no rebasara los 15 saltos máximos de la métrica por lo que el número máximo de saltos que registramos entre los nodos de la red fueron entre Macedonia-Finlandia con un máximo de 8 saltos como se muestra en la figura 92.

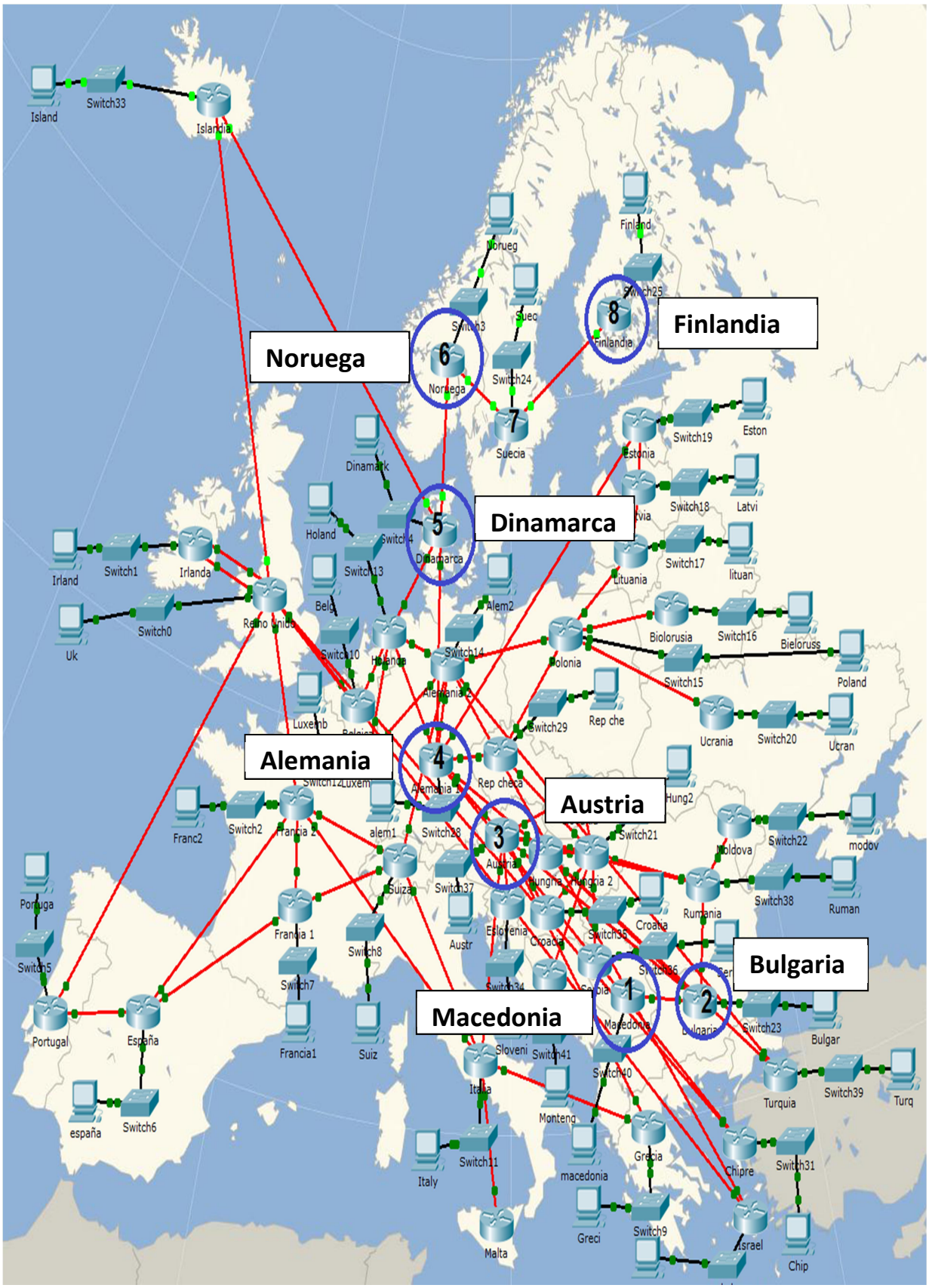


Figura 92. Conteo de saltos del PING de Macedonia a Finlandia

5.1.7. Tablas de enrutamiento de RIP v2 en los routers de la simulación de GEANT

En esta sección demostramos la buena configuración y funcionamiento del protocolo RIP v2, como evidencia se pueden analizar las tablas de enrutamiento de todos los routers de la red y todos deben de contener las mismas rutas de red, en este caso analizaremos al azar las tablas del router de UK y Ucrania como se muestran en la figura 93, 93.1 y 93.2.

```
R    10.0.0.0/8 [120/2] via 194.10.2.4, 00:00:27, GigabitEthernet3/0
R    12.0.0.0/8 [120/3] via 194.10.2.4, 00:00:27, GigabitEthernet3/0
                [120/3] via 197.10.2.4, 00:00:25, GigabitEthernet6/0
R    14.0.0.0/8 [120/3] via 194.10.2.4, 00:00:27, GigabitEthernet3/0
                [120/3] via 197.10.2.4, 00:00:25, GigabitEthernet6/0
R    15.0.0.0/8 [120/3] via 194.10.2.4, 00:00:27, GigabitEthernet3/0
                [120/3] via 197.10.2.4, 00:00:25, GigabitEthernet6/0
R    16.0.0.0/8 [120/3] via 194.10.2.4, 00:00:27, GigabitEthernet3/0
                [120/3] via 197.10.2.4, 00:00:25, GigabitEthernet6/0
R    17.0.0.0/8 [120/3] via 194.10.2.4, 00:00:27, GigabitEthernet3/0
                [120/3] via 197.10.2.4, 00:00:25, GigabitEthernet6/0
R    18.0.0.0/8 [120/3] via 194.10.2.4, 00:00:27, GigabitEthernet3/0
                [120/3] via 197.10.2.4, 00:00:25, GigabitEthernet6/0
R    22.0.0.0/8 [120/1] via 199.10.2.4, 00:00:23, GigabitEthernet8/0
R    40.0.0.0/8 [120/1] via 195.10.2.4, 00:00:23, GigabitEthernet4/0
R    41.0.0.0/8 [120/1] via 194.10.2.4, 00:00:27, GigabitEthernet3/0
R    42.0.0.0/8 [120/1] via 194.10.2.4, 00:00:27, GigabitEthernet3/0
R    43.0.0.0/8 [120/2] via 194.10.2.4, 00:00:27, GigabitEthernet3/0
                [120/2] via 195.10.2.4, 00:00:23, GigabitEthernet4/0
R    44.0.0.0/8 [120/2] via 194.10.2.4, 00:00:27, GigabitEthernet3/0
R    45.0.0.0/8 [120/1] via 194.10.2.4, 00:00:27, GigabitEthernet3/0
R    46.0.0.0/8 [120/1] via 194.10.2.4, 00:00:27, GigabitEthernet3/0
R    47.0.0.0/8 [120/2] via 194.10.2.4, 00:00:27, GigabitEthernet3/0
R    48.0.0.0/8 [120/2] via 194.10.2.4, 00:00:27, GigabitEthernet3/0
R    49.0.0.0/8 [120/2] via 194.10.2.4, 00:00:27, GigabitEthernet3/0
R    51.0.0.0/8 [120/3] via 197.10.2.4, 00:00:25, GigabitEthernet6/0
                [120/3] via 198.10.2.4, 00:00:24, GigabitEthernet7/0
R    52.0.0.0/8 [120/4] via 194.10.2.4, 00:00:27, GigabitEthernet3/0

R    53.0.0.0/8 [120/4] via 197.10.2.4, 00:00:25, GigabitEthernet6/0
                [120/4] via 194.10.2.4, 00:00:27, GigabitEthernet3/0
R    54.0.0.0/8 [120/4] via 197.10.2.4, 00:00:25, GigabitEthernet6/0
                [120/4] via 194.10.2.4, 00:00:27, GigabitEthernet3/0
R    90.0.0.0/8 [120/3] via 198.10.2.4, 00:00:24, GigabitEthernet7/0
R    91.0.0.0/8 [120/3] via 198.10.2.4, 00:00:24, GigabitEthernet7/0
R    92.0.0.0/8 [120/3] via 198.10.2.4, 00:00:24, GigabitEthernet7/0
R    93.0.0.0/8 [120/4] via 197.10.2.4, 00:00:25, GigabitEthernet6/0
                [120/4] via 198.10.2.4, 00:00:24, GigabitEthernet7/0
R    94.0.0.0/8 [120/3] via 197.10.2.4, 00:00:25, GigabitEthernet6/0
R    95.0.0.0/8 [120/2] via 199.10.2.4, 00:00:23, GigabitEthernet8/0
R    96.0.0.0/8 [120/3] via 199.10.2.4, 00:00:23, GigabitEthernet8/0
R    98.0.0.0/8 [120/2] via 196.10.2.4, 00:00:01, GigabitEthernet5/0
                [120/2] via 199.10.2.4, 00:00:23, GigabitEthernet8/0
R    99.0.0.0/8 [120/1] via 196.10.2.4, 00:00:01, GigabitEthernet5/0
R    100.0.0.0/8 [120/2] via 196.10.2.4, 00:00:01, GigabitEthernet5/0
R    101.0.0.0/8 [120/4] via 199.10.2.4, 00:00:23, GigabitEthernet8/0
R    129.10.0.0/16 [120/2] via 197.10.2.4, 00:00:25, GigabitEthernet6/0
                   [120/2] via 198.10.2.4, 00:00:24, GigabitEthernet7/0
R    130.10.0.0/16 [120/2] via 196.10.2.4, 00:00:01, GigabitEthernet5/0
                   [120/2] via 198.10.2.4, 00:00:24, GigabitEthernet7/0
R    131.10.0.0/16 [120/2] via 198.10.2.4, 00:00:24, GigabitEthernet7/0
R    132.10.0.0/16 [120/2] via 198.10.2.4, 00:00:24, GigabitEthernet7/0
R    133.10.0.0/16 [120/2] via 194.10.2.4, 00:00:27, GigabitEthernet3/0
                   [120/2] via 198.10.2.4, 00:00:24, GigabitEthernet7/0
```

Figura 93. Tabla de enrutamiento de UK parte 1.

```
R    134.10.0.0/16 [120/1] via 198.10.2.4, 00:00:24, GigabitEthernet7/0
R    135.10.0.0/16 [120/2] via 198.10.2.4, 00:00:24, GigabitEthernet7/0
R    136.10.0.0/16 [120/2] via 197.10.2.4, 00:00:25, GigabitEthernet6/0
R    137.10.0.0/16 [120/2] via 197.10.2.4, 00:00:25, GigabitEthernet6/0
R    138.10.0.0/16 [120/2] via 197.10.2.4, 00:00:25, GigabitEthernet6/0
R    139.10.0.0/16 [120/1] via 197.10.2.4, 00:00:25, GigabitEthernet6/0
R    140.10.0.0/16 [120/2] via 196.10.2.4, 00:00:01, GigabitEthernet5/0
                   [120/2] via 197.10.2.4, 00:00:25, GigabitEthernet6/0
R    141.10.0.0/16 [120/2] via 197.10.2.4, 00:00:25, GigabitEthernet6/0
R    142.10.0.0/16 [120/2] via 199.10.2.4, 00:00:23, GigabitEthernet8/0
                   [120/2] via 197.10.2.4, 00:00:25, GigabitEthernet6/0
R    150.10.0.0/16 [120/3] via 197.10.2.4, 00:00:25, GigabitEthernet6/0
                   [120/3] via 194.10.2.4, 00:00:27, GigabitEthernet3/0
R    151.10.0.0/16 [120/4] via 197.10.2.4, 00:00:25, GigabitEthernet6/0
R    152.10.0.0/16 [120/4] via 194.10.2.4, 00:00:27, GigabitEthernet3/0
                   [120/4] via 197.10.2.4, 00:00:25, GigabitEthernet6/0
R    153.10.0.0/16 [120/4] via 194.10.2.4, 00:00:27, GigabitEthernet3/0
                   [120/4] via 197.10.2.4, 00:00:25, GigabitEthernet6/0
R    154.10.0.0/16 [120/3] via 197.10.2.4, 00:00:25, GigabitEthernet6/0
R    155.10.0.0/16 [120/4] via 197.10.2.4, 00:00:25, GigabitEthernet6/0
R    156.10.0.0/16 [120/4] via 197.10.2.4, 00:00:25, GigabitEthernet6/0
                   [120/4] via 194.10.2.4, 00:00:27, GigabitEthernet3/0
R    157.10.0.0/16 [120/3] via 197.10.2.4, 00:00:25, GigabitEthernet6/0
R    158.10.0.0/16 [120/3] via 198.10.2.4, 00:00:24, GigabitEthernet7/0
                   [120/3] via 197.10.2.4, 00:00:25, GigabitEthernet6/0
R    159.10.0.0/16 [120/3] via 197.10.2.4, 00:00:25, GigabitEthernet6/0
C    192.10.2.0/24 is directly connected, GigabitEthernet1/0
R    192.168.2.0/24 [120/2] via 198.10.2.4, 00:00:24, GigabitEthernet7/0
R    192.168.3.0/24 [120/2] via 196.10.2.4, 00:00:01, GigabitEthernet5/0
R    192.168.4.0/24 [120/2] via 197.10.2.4, 00:00:25, GigabitEthernet6/0
R    192.168.5.0/24 [120/3] via 198.10.2.4, 00:00:24, GigabitEthernet7/0
R    192.168.6.0/24 [120/3] via 196.10.2.4, 00:00:01, GigabitEthernet5/0

R    192.168.7.0/24 [120/2] via 194.10.2.4, 00:00:27, GigabitEthernet3/0
R    192.168.8.0/24 [120/1] via 198.10.2.4, 00:00:24, GigabitEthernet7/0
R    192.168.9.0/24 [120/3] via 198.10.2.4, 00:00:24, GigabitEthernet7/0
R    192.168.10.0/24 [120/3] via 194.10.2.4, 00:00:27, GigabitEthernet3/0
                     [120/3] via 197.10.2.4, 00:00:25, GigabitEthernet6/0
R    192.168.12.0/24 [120/3] via 197.10.2.4, 00:00:25, GigabitEthernet6/0
R    192.168.13.0/24 [120/1] via 197.10.2.4, 00:00:25, GigabitEthernet6/0
R    192.168.14.0/24 [120/3] via 197.10.2.4, 00:00:25, GigabitEthernet6/0
R    192.168.15.0/24 [120/2] via 199.10.2.4, 00:00:23, GigabitEthernet8/0
R    192.168.17.0/24 [120/4] via 197.10.2.4, 00:00:25, GigabitEthernet6/0
R    192.168.18.0/24 [120/5] via 197.10.2.4, 00:00:25, GigabitEthernet6/0
R    192.168.19.0/24 [120/4] via 197.10.2.4, 00:00:25, GigabitEthernet6/0
                     [120/4] via 194.10.2.4, 00:00:27, GigabitEthernet3/0
R    192.168.20.0/24 [120/4] via 194.10.2.4, 00:00:27, GigabitEthernet3/0
                     [120/4] via 197.10.2.4, 00:00:25, GigabitEthernet6/0
R    192.168.21.0/24 [120/5] via 197.10.2.4, 00:00:25, GigabitEthernet6/0
R    192.168.22.0/24 [120/4] via 197.10.2.4, 00:00:25, GigabitEthernet6/0
                     [120/4] via 194.10.2.4, 00:00:27, GigabitEthernet3/0
R    192.168.24.0/24 [120/4] via 197.10.2.4, 00:00:25, GigabitEthernet6/0
                     [120/4] via 194.10.2.4, 00:00:27, GigabitEthernet3/0
R    192.168.31.0/24 [120/2] via 194.10.2.4, 00:00:27, GigabitEthernet3/0
R    192.168.32.0/24 [120/3] via 194.10.2.4, 00:00:27, GigabitEthernet3/0
R    192.168.33.0/24 [120/4] via 194.10.2.4, 00:00:27, GigabitEthernet3/0
                     [120/4] via 197.10.2.4, 00:00:25, GigabitEthernet6/0
R    192.168.35.0/24 [120/1] via 193.10.2.4, 00:00:27, GigabitEthernet2/0
                     [120/1] via 192.10.2.4, 00:00:27, GigabitEthernet1/0
C    192.168.36.0/24 is directly connected, FastEthernet0/0
R    192.168.37.0/24 [120/2] via 194.10.2.4, 00:00:27, GigabitEthernet3/0
R    192.168.38.0/24 [120/1] via 195.10.2.4, 00:00:23, GigabitEthernet4/0
R    192.168.39.0/24 [120/1] via 196.10.2.4, 00:00:01, GigabitEthernet5/0
R    192.168.40.0/24 [120/1] via 199.10.2.4, 00:00:23, GigabitEthernet8/0
R    192.168.43.0/24 [120/2] via 194.10.2.4, 00:00:27, GigabitEthernet3/0
                     [120/2] via 195.10.2.4, 00:00:23, GigabitEthernet4/0

R    192.168.44.0/24 [120/1] via 194.10.2.4, 00:00:27, GigabitEthernet3/0
R    192.168.46.0/24 [120/3] via 194.10.2.4, 00:00:27, GigabitEthernet3/0
R    192.168.53.0/24 [120/5] via 197.10.2.4, 00:00:25, GigabitEthernet6/0
                     [120/5] via 194.10.2.4, 00:00:27, GigabitEthernet3/0
R    192.168.67.0/24 [120/5] via 197.10.2.4, 00:00:25, GigabitEthernet6/0
                     [120/5] via 194.10.2.4, 00:00:27, GigabitEthernet3/0
R    192.168.73.0/24 [120/4] via 198.10.2.4, 00:00:24, GigabitEthernet7/0
```

Figura 93.1. Tabla de enrutamiento de UK parte 2.

```
R    192.168.73.0/24 [120/4] via 198.10.2.4, 00:00:24, GigabitEthernet7/0
R    192.168.74.0/24 [120/4] via 198.10.2.4, 00:00:24, GigabitEthernet7/0
R    192.168.75.0/24 [120/4] via 198.10.2.4, 00:00:24, GigabitEthernet7/0
R    192.168.77.0/24 [120/4] via 197.10.2.4, 00:00:25, GigabitEthernet6/0
R    192.168.78.0/24 [120/3] via 199.10.2.4, 00:00:23, GigabitEthernet8/0
R    192.168.79.0/24 [120/4] via 199.10.2.4, 00:00:23, GigabitEthernet8/0
R    192.168.80.0/24 [120/5] via 199.10.2.4, 00:00:23, GigabitEthernet8/0
C    193.10.2.0/24 is directly connected, GigabitEthernet2/0
C    194.10.2.0/24 is directly connected, GigabitEthernet3/0
C    195.10.2.0/24 is directly connected, GigabitEthernet4/0
C    196.10.2.0/24 is directly connected, GigabitEthernet5/0
C    197.10.2.0/24 is directly connected, GigabitEthernet6/0
C    198.10.2.0/24 is directly connected, GigabitEthernet7/0
C    199.10.2.0/24 is directly connected, GigabitEthernet8/0
```

Figura 93.2. Tabla de enrutamiento de UK parte 3.

Ahora en comparación, mostraremos la tabla de enrutamiento del nodo de Ucrania, la cual solo tiene un punto de acceso hacia el router de Polonia. Sin embargo, deben contener las mismas direcciones de red de la tabla de enrutamiento del UK, como se muestra en la figuras 94-94.1.

```
R     10.0.0.0/8 [120/4] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R     12.0.0.0/8 [120/4] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R     14.0.0.0/8 [120/4] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R     15.0.0.0/8 [120/4] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R     16.0.0.0/8 [120/4] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R     17.0.0.0/8 [120/4] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R     18.0.0.0/8 [120/4] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R     22.0.0.0/8 [120/4] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R     40.0.0.0/8 [120/5] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R     41.0.0.0/8 [120/4] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R     42.0.0.0/8 [120/4] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R     43.0.0.0/8 [120/4] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R     44.0.0.0/8 [120/3] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R     45.0.0.0/8 [120/3] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R     46.0.0.0/8 [120/4] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R     47.0.0.0/8 [120/3] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R     48.0.0.0/8 [120/4] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R     49.0.0.0/8 [120/4] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R     51.0.0.0/8 [120/1] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R     52.0.0.0/8 [120/5] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R     53.0.0.0/8 [120/4] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R     54.0.0.0/8 [120/4] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R     90.0.0.0/8 [120/1] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R     91.0.0.0/8 [120/1] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
C     92.0.0.0/8 is directly connected, GigabitEthernet4/0
R     93.0.0.0/8 [120/2] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R     94.0.0.0/8 [120/3] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R     95.0.0.0/8 [120/4] via 92.10.2.3, 00:00:16, GigabitEthernet4/0

R     96.0.0.0/8 [120/5] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R     98.0.0.0/8 [120/3] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R     99.0.0.0/8 [120/3] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R     100.0.0.0/8 [120/3] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R     101.0.0.0/8 [120/6] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R     129.10.0.0/16 [120/2] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R     130.10.0.0/16 [120/2] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R     131.10.0.0/16 [120/1] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R     132.10.0.0/16 [120/2] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R     133.10.0.0/16 [120/2] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R     134.10.0.0/16 [120/2] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R     135.10.0.0/16 [120/2] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R     136.10.0.0/16 [120/3] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R     137.10.0.0/16 [120/3] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
```

Figura 94. Tabla de enrutamiento de Ucrania, parte 1.

```
R    137.10.0.0/16 [120/3] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R    138.10.0.0/16 [120/2] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R    139.10.0.0/16 [120/3] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R    140.10.0.0/16 [120/3] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R    141.10.0.0/16 [120/3] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R    142.10.0.0/16 [120/3] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R    150.10.0.0/16 [120/3] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R    151.10.0.0/16 [120/3] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R    152.10.0.0/16 [120/3] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R    153.10.0.0/16 [120/3] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R    154.10.0.0/16 [120/2] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R    155.10.0.0/16 [120/3] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R    156.10.0.0/16 [120/3] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R    157.10.0.0/16 [120/4] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R    158.10.0.0/16 [120/3] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R    159.10.0.0/16 [120/3] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R    192.10.2.0/24 [120/4] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R    192.168.2.0/24 [120/2] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R    192.168.3.0/24 [120/3] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R    192.168.4.0/24 [120/3] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R    192.168.5.0/24 [120/1] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R    192.168.6.0/24 [120/3] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R    192.168.7.0/24 [120/3] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R    192.168.8.0/24 [120/3] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R    192.168.9.0/24 [120/3] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R    192.168.10.0/24 [120/4] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R    192.168.12.0/24 [120/2] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R    192.168.13.0/24 [120/4] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R    192.168.14.0/24 [120/4] via 92.10.2.3, 00:00:16, GigabitEthernet4/0

R    192.168.15.0/24 [120/4] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R    192.168.17.0/24 [120/3] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R    192.168.18.0/24 [120/4] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R    192.168.19.0/24 [120/4] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R    192.168.20.0/24 [120/4] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R    192.168.21.0/24 [120/4] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R    192.168.22.0/24 [120/4] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R    192.168.24.0/24 [120/5] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R    192.168.31.0/24 [120/4] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R    192.168.32.0/24 [120/5] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R    192.168.33.0/24 [120/5] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R    192.168.35.0/24 [120/5] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R    192.168.36.0/24 [120/4] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R    192.168.37.0/24 [120/4] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R    192.168.38.0/24 [120/5] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R    192.168.39.0/24 [120/4] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R    192.168.40.0/24 [120/5] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R    192.168.43.0/24 [120/5] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R    192.168.44.0/24 [120/4] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R    192.168.46.0/24 [120/5] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R    192.168.53.0/24 [120/6] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R    192.168.67.0/24 [120/5] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R    192.168.73.0/24 [120/2] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R    192.168.74.0/24 [120/2] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
C    192.168.75.0/24 is directly connected, FastEthernet0/0
R    192.168.77.0/24 [120/3] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R    192.168.78.0/24 [120/5] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R    192.168.79.0/24 [120/6] via 92.10.2.3, 00:00:16, GigabitEthernet4/0

R    192.168.80.0/24 [120/7] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R    193.10.2.0/24 [120/4] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R    194.10.2.0/24 [120/4] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R    195.10.2.0/24 [120/4] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R    196.10.2.0/24 [120/4] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R    197.10.2.0/24 [120/4] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R    198.10.2.0/24 [120/3] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
R    199.10.2.0/24 [120/4] via 92.10.2.3, 00:00:16, GigabitEthernet4/0
```

Figura 94.1. Tabla de enrutamiento del router de Ucrania, parte 2.

Las pequeñas variaciones que observamos en las tablas de enrutamiento, son por las redes que se conectan directamente en cada router y son identificadas con el prefijo inicial “C”, mientras que las direcciones con prefijo “R” nos indican que fueron configuradas con el protocolo RIP. En caso de

querer comprobar las interfaces de red configuradas, ir a las tablas de red de la simulación mostradas en las páginas 66-68.

5.1.8. Prueba de Latencia en la simulación de GEANT con RIP v2

En esta etapa pondremos a prueba el protocolo RIP v2 al mantener pruebas de conectividad simultaneas entre los nodos de UK-Bélgica, Alemania-España, Suiza-Turquía, Francia 1-Ucrania, Austria-Irlanda, Hungria-Letonia y Noruega-Estonia, deshabilitando el PoP de Alemania para poner a prueba la convergencia y latencia de los mensajes ante un evento inesperado como la caída de un router Backbone en la red. Como evidencia también medimos el rendimiento del equipo físico ante el funcionamiento total de la simulación en Packet Tracer, con un uso promedio de CPU de 10.4 %y 2.0025 % de RAM como se muestra en la figura 95.

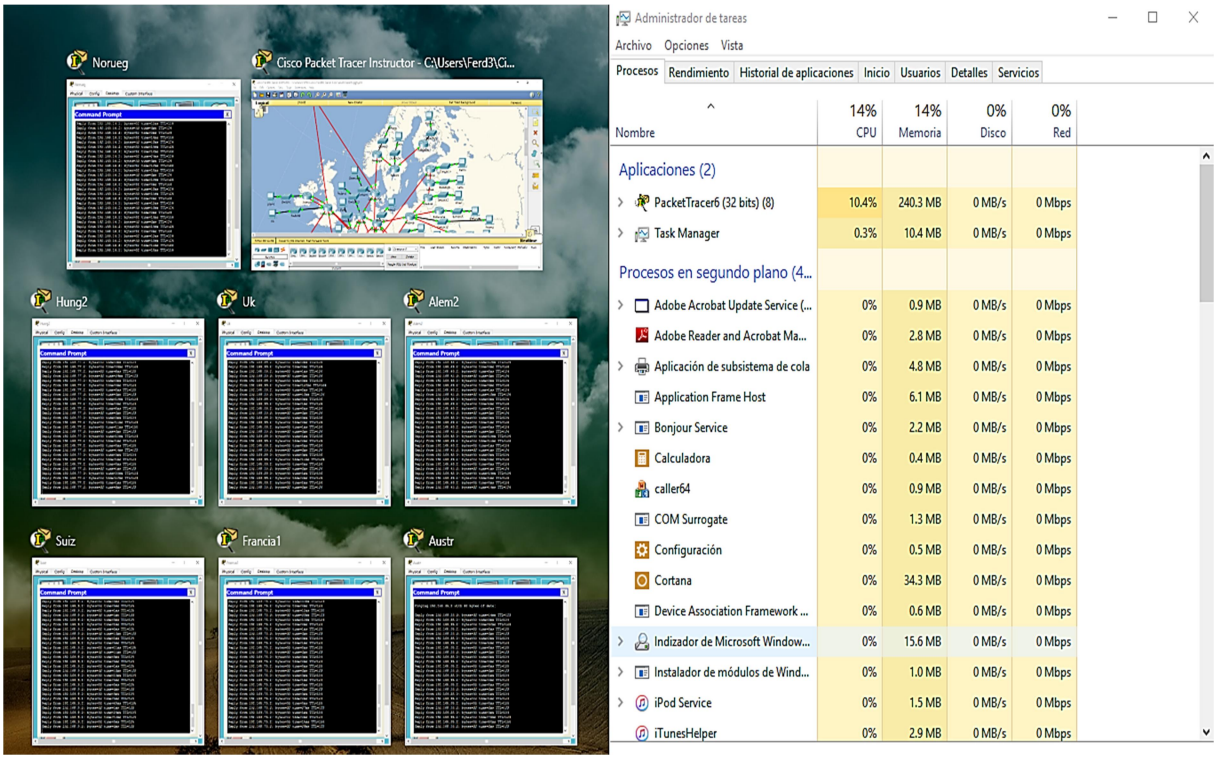


Figura.95. Rendimiento en simulación s simultánea de la red GEANT.

Al realizar la pruebas simultaneas de conectividad entre los países mencionados y deshabilitando el router de Alemania, comprobamos que no hubo desconexión total en la red y en algunos casos solo hubo retrasos en la entrega de paquetes como se muestra en las figuras 96-102.

```
Reply from 192.168.39.2: bytes=32 time=0ms TTL=126
Reply from 192.168.39.2: bytes=32 time=33ms TTL=126
Reply from 192.168.39.2: bytes=32 time=992ms TTL=126
Reply from 192.168.39.2: bytes=32 time=4541ms TTL=126
Request timed out.
Reply from 192.168.39.2: bytes=32 time=2986ms TTL=126
Reply from 192.168.39.2: bytes=32 time=1262ms TTL=126
Reply from 192.168.39.2: bytes=32 time=109ms TTL=126
```

Figura 96. Prueba de Conectividad simultánea entre UK-Bélgica.

```
Reply from 192.168.43.2: bytes=32 time=104ms TTL=124
Reply from 192.168.43.2: bytes=32 time=0ms TTL=124
Reply from 192.168.43.2: bytes=32 time=4648ms TTL=124
Request timed out.
Request timed out.
Reply from 192.168.43.2: bytes=32 time=38ms TTL=124
Request timed out.
Reply from 192.168.43.2: bytes=32 time=4714ms TTL=124
Reply from 192.168.43.2: bytes=32 time=4713ms TTL=124
```

Figura 97. Prueba de Conectividad simultánea entre Alemania-España.

```
Reply from 192.168.9.2: bytes=32 time=0ms TTL=125
Reply from 133.10.2.3: Destination host unreachable
Reply from 133.10.2.3: Destination host unreachable
Reply from 133.10.2.3: Destination host unreachable
Request timed out.
Request timed out.
Request timed out.
Reply from 192.168.9.2: bytes=32 time=1933ms TTL=122
Reply from 192.168.9.2: bytes=32 time=2126ms TTL=122
```

Figura 98. Prueba de Conectividad simultánea entre Suiza-Turquía.

```
Reply from 192.168.75.2: bytes=32 time=99ms TTL=123
Reply from 133.10.2.3: Destination host unreachable
Reply from 192.168.75.2: bytes=32 time=10ms TTL=119
Reply from 192.168.75.2: bytes=32 time=195ms TTL=120
Reply from 192.168.75.2: bytes=32 time=10ms TTL=120
Reply from 192.168.75.2: bytes=32 time=168ms TTL=120
Reply from 192.168.75.2: bytes=32 time=14ms TTL=120
Request timed out.
Request timed out.
Request timed out.
Request timed out.
Reply from 192.168.75.2: bytes=32 time=2825ms TTL=120
Reply from 192.168.75.2: bytes=32 time=2866ms TTL=120
```

Figura 99. Prueba de Conectividad simultánea entre Francia 1-Ucrania.

```
Reply from 192.168.35.2: bytes=32 time=17ms TTL=123
Reply from 192.168.35.2: bytes=32 time=138ms TTL=123
Reply from 192.168.35.2: bytes=32 time=12ms TTL=123
Request timed out.
Request timed out.
Request timed out.
Request timed out.
Reply from 192.168.35.2: bytes=32 time=2840ms TTL=123
Reply from 192.168.35.2: bytes=32 time=3217ms TTL=123
```

Figura 100. Prueba de Conectividad simultánea entre Austria-Irlanda.

```
Reply from 192.168.77.2: bytes=32 time=21ms TTL=123
Reply from 192.168.77.2: bytes=32 time=2ms TTL=123
Reply from 192.168.77.2: bytes=32 time=1ms TTL=123
Request timed out.
Reply from 192.168.77.2: bytes=32 time=2390ms TTL=123
Request timed out.
Reply from 192.168.77.2: bytes=32 time=2906ms TTL=123
Request timed out.
Reply from 192.168.77.2: bytes=32 time=1205ms TTL=123
Reply from 192.168.77.2: bytes=32 time=1392ms TTL=123
```

Figura 101. Prueba de Conectividad simultánea entre Hungria-Letonia

```
Reply from 192.168.14.2: bytes=32 time=1ms TTL=124
Reply from 192.168.14.2: bytes=32 time=401ms TTL=124
Request timed out.
Request timed out.
Request timed out.
Reply from 192.168.14.2: bytes=32 time=3395ms TTL=124
Reply from 192.168.14.2: bytes=32 time=4294967113ms TTL=124
Reply from 192.168.14.2: bytes=32 time=2546ms TTL=124
Reply from 192.168.14.2: bytes=32 time=3522ms TTL=124
```

Figura 102. Prueba de Conectividad simultánea entre Noruega-Estonia.

Como observamos en las figuras anteriores, el tiempo de retardo en cada uno de los paquetes es variado y en algunos casos muy extenso, con tiempos de retardo superiores a 2000 milisegundos por paquete, sin embargo, el tiempo que toma el algoritmo en actualizar las rutas y redirigir el tráfico genero una pérdida de 3 paquetes perdidos por prueba sin perder en su totalidad conectividad en ninguno de los casos.

5.1.9. Prueba de conectividad con el protocolo OSPF en simulación

En esta sección probaremos conectividad de la red como lo hicimos con el protocolo RIP v2, realizando pruebas PING y comprobando las tablas de enrutamiento. A continuación mostramos algunas pruebas de conectividad desde el PoP de Alemania 2 hacia el resto de la red. Ver figuras 103-105.

```
PC>ping 192.168.10.2

Pinging 192.168.10.2 with 32 bytes of data:

Request timed out.
Reply from 192.168.10.2: bytes=32 time=10ms TTL=124
Reply from 192.168.10.2: bytes=32 time=11ms TTL=124
Reply from 192.168.10.2: bytes=32 time=10ms TTL=124

Ping statistics for 192.168.10.2:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 10ms, Maximum = 11ms, Average = 10ms

PC>ping 192.168.39.2

Pinging 192.168.39.2 with 32 bytes of data:

Request timed out.
Reply from 192.168.39.2: bytes=32 time=13ms TTL=125
Reply from 192.168.39.2: bytes=32 time=13ms TTL=125
Reply from 192.168.39.2: bytes=32 time=12ms TTL=125

Ping statistics for 192.168.39.2:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 12ms, Maximum = 13ms, Average = 12ms
```

Figura 103. Ping de Alemania a Australia y Bélgica.

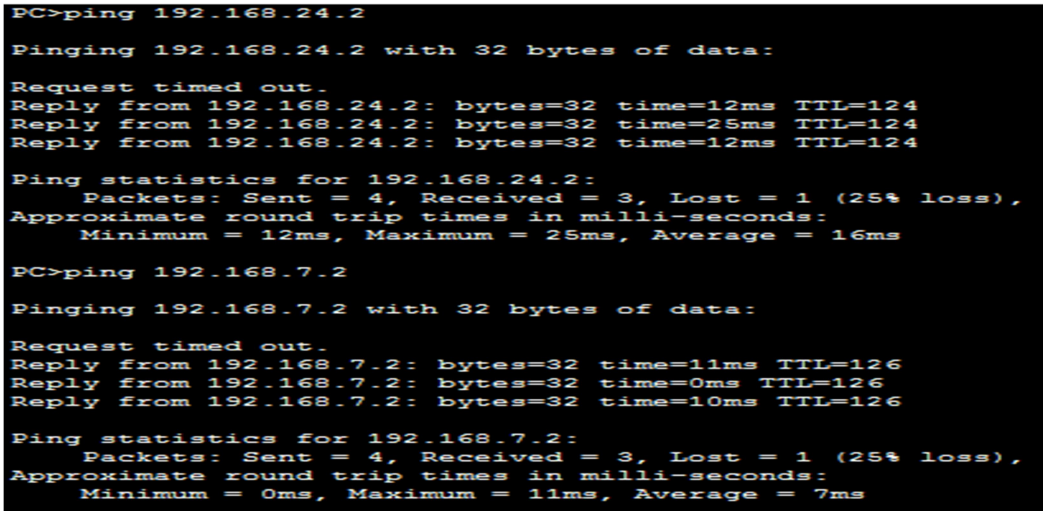


Figura 104. Prueba de conectividad, de Alemania 2 a Bulgaria y Suiza

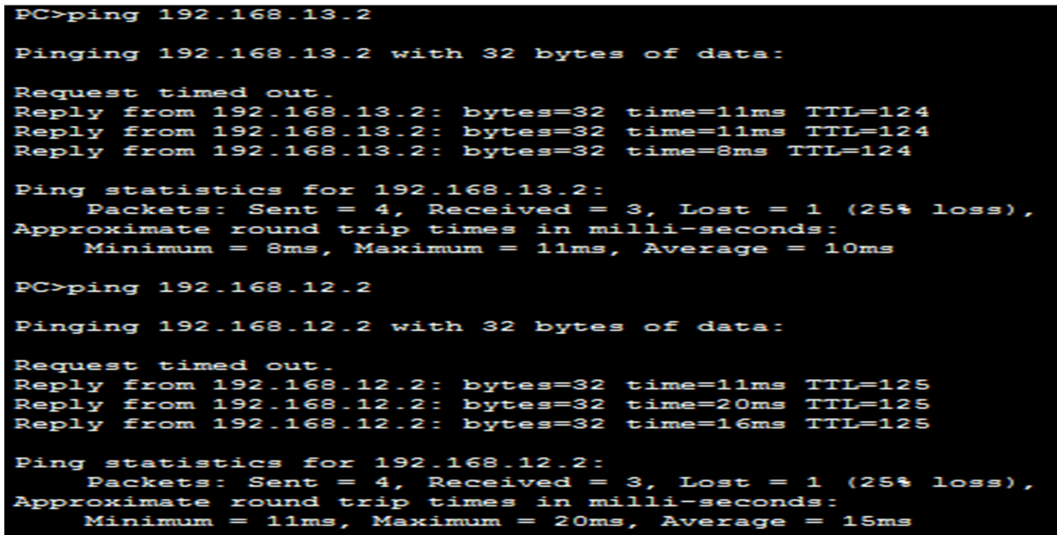


Figura 105. Prueba de conectividad de Alemania 2 a Chipre y Rep. Checa.

El resto de las evidencias de conectividad hacia el resto de los nodos de la red se realizaron con éxito y se encuentran en el apéndice B.

5.1.10. Tablas de Enrutamiento con el Protocolo OSPF

Como demostramos anteriormente las tablas de enrutamiento, nos indican un mapa de las direcciones de la red y cada router contiene una tabla con las mismas direcciones. A diferencia de las tablas con RIP estas tablas están identificadas con el prefijo “O” el cual indica que están configuradas con el protocolo OSPF, también observamos que algunas redes están señalizadas con el prefijo “IA” (*Intra-Area*), lo que significa que estas redes se conectan con otras áreas, como se muestra en la tabla de enrutamiento del router de Suiza, en las figuras 106-106.1.

```
O IA 10.0.0.0/8 [110/5] via 133.10.2.3, 00:03:57, GigabitEthernet5/0
O IA 12.0.0.0/8 [110/5] via 133.10.2.3, 00:03:57, GigabitEthernet5/0
O IA 14.0.0.0/8 [110/5] via 133.10.2.3, 00:03:57, GigabitEthernet5/0
O IA 15.0.0.0/8 [110/5] via 133.10.2.3, 00:03:57, GigabitEthernet5/0
O IA 16.0.0.0/8 [110/5] via 133.10.2.3, 00:03:57, GigabitEthernet5/0
O IA 17.0.0.0/8 [110/5] via 133.10.2.3, 00:03:57, GigabitEthernet5/0
O IA 18.0.0.0/8 [110/5] via 133.10.2.3, 00:03:57, GigabitEthernet5/0
O IA 22.0.0.0/8 [110/4] via 45.10.2.3, 00:04:24, GigabitEthernet4/0
                [110/4] via 133.10.2.3, 00:03:57, GigabitEthernet5/0
O    40.0.0.0/8 [110/3] via 44.10.2.3, 00:04:34, GigabitEthernet7/0
                [110/3] via 45.10.2.3, 00:04:34, GigabitEthernet4/0
O    41.0.0.0/8 [110/2] via 45.10.2.3, 00:04:34, GigabitEthernet4/0
O    42.0.0.0/8 [110/2] via 44.10.2.3, 00:04:34, GigabitEthernet7/0
                [110/2] via 45.10.2.3, 00:04:34, GigabitEthernet4/0
O    43.0.0.0/8 [110/2] via 44.10.2.3, 00:04:34, GigabitEthernet7/0
C    44.0.0.0/8 is directly connected, GigabitEthernet7/0
C    45.0.0.0/8 is directly connected, GigabitEthernet4/0
O    46.0.0.0/8 [110/2] via 45.10.2.3, 00:04:34, GigabitEthernet4/0
                [110/2] via 47.10.2.4, 00:04:34, GigabitEthernet1/0
C    47.0.0.0/8 is directly connected, GigabitEthernet1/0
O    48.0.0.0/8 [110/2] via 47.10.2.4, 00:04:34, GigabitEthernet1/0
O    49.0.0.0/8 [110/2] via 47.10.2.4, 00:04:34, GigabitEthernet1/0
O IA 51.0.0.0/8 [110/3] via 133.10.2.3, 00:04:34, GigabitEthernet5/0
O IA 52.0.0.0/8 [110/5] via 133.10.2.3, 00:03:57, GigabitEthernet5/0
O IA 53.0.0.0/8 [110/5] via 133.10.2.3, 00:03:57, GigabitEthernet5/0
O IA 54.0.0.0/8 [110/6] via 133.10.2.3, 00:04:34, GigabitEthernet5/0
O IA 90.0.0.0/8 [110/3] via 133.10.2.3, 00:04:34, GigabitEthernet5/0
O IA 91.0.0.0/8 [110/3] via 133.10.2.3, 00:04:34, GigabitEthernet5/0
O IA 92.0.0.0/8 [110/3] via 133.10.2.3, 00:04:34, GigabitEthernet5/0
O IA 93.0.0.0/8 [110/6] via 133.10.2.3, 00:03:57, GigabitEthernet5/0
O IA 94.0.0.0/8 [110/5] via 133.10.2.3, 00:03:57, GigabitEthernet5/0
O IA 95.0.0.0/8 [110/4] via 133.10.2.3, 00:03:57, GigabitEthernet5/0
O IA 96.0.0.0/8 [110/5] via 133.10.2.3, 00:03:57, GigabitEthernet5/0

O IA 98.0.0.0/8 [110/3] via 133.10.2.3, 00:03:57, GigabitEthernet5/0
O IA 99.0.0.0/8 [110/3] via 133.10.2.3, 00:03:57, GigabitEthernet5/0
O IA 100.0.0.0/8 [110/3] via 133.10.2.3, 00:03:57, GigabitEthernet5/0
O IA 101.0.0.0/8 [110/6] via 133.10.2.3, 00:03:57, GigabitEthernet5/0
O IA 129.10.0.0/16 [110/2] via 133.10.2.3, 00:04:34, GigabitEthernet5/0
O IA 130.10.0.0/16 [110/2] via 133.10.2.3, 00:04:34, GigabitEthernet5/0
O IA 131.10.0.0/16 [110/2] via 133.10.2.3, 00:04:34, GigabitEthernet5/0
O IA 132.10.0.0/16 [110/2] via 133.10.2.3, 00:04:34, GigabitEthernet5/0
C    133.10.0.0/16 is directly connected, GigabitEthernet5/0
O IA 134.10.0.0/16 [110/2] via 133.10.2.3, 00:04:34, GigabitEthernet5/0
O IA 135.10.0.0/16 [110/2] via 133.10.2.3, 00:04:34, GigabitEthernet5/0
O IA 136.10.0.0/16 [110/4] via 133.10.2.3, 00:03:57, GigabitEthernet5/0
O IA 137.10.0.0/16 [110/4] via 133.10.2.3, 00:03:57, GigabitEthernet5/0
O IA 138.10.0.0/16 [110/4] via 133.10.2.3, 00:04:34, GigabitEthernet5/0
O IA 139.10.0.0/16 [110/4] via 133.10.2.3, 00:03:57, GigabitEthernet5/0
                   [110/4] via 45.10.2.3, 00:03:57, GigabitEthernet4/0
O IA 140.10.0.0/16 [110/3] via 133.10.2.3, 00:03:57, GigabitEthernet5/0
O IA 141.10.0.0/16 [110/4] via 133.10.2.3, 00:03:57, GigabitEthernet5/0
O IA 142.10.0.0/16 [110/4] via 133.10.2.3, 00:03:57, GigabitEthernet5/0
O IA 150.10.0.0/16 [110/5] via 133.10.2.3, 00:04:34, GigabitEthernet5/0
O IA 151.10.0.0/16 [110/5] via 133.10.2.3, 00:04:34, GigabitEthernet5/0
O IA 152.10.0.0/16 [110/5] via 133.10.2.3, 00:04:34, GigabitEthernet5/0
O IA 153.10.0.0/16 [110/5] via 133.10.2.3, 00:04:34, GigabitEthernet5/0
O IA 154.10.0.0/16 [110/4] via 133.10.2.3, 00:04:34, GigabitEthernet5/0
O IA 155.10.0.0/16 [110/5] via 133.10.2.3, 00:04:34, GigabitEthernet5/0
O IA 156.10.0.0/16 [110/5] via 133.10.2.3, 00:04:34, GigabitEthernet5/0
O IA 157.10.0.0/16 [110/4] via 133.10.2.3, 00:03:57, GigabitEthernet5/0
O IA 158.10.0.0/16 [110/3] via 133.10.2.3, 00:03:57, GigabitEthernet5/0
O IA 159.10.0.0/16 [110/4] via 133.10.2.3, 00:03:57, GigabitEthernet5/0
O    192.10.2.0/24 [110/3] via 45.10.2.3, 00:04:34, GigabitEthernet4/0
O IA 192.168.2.0/24 [110/2] via 133.10.2.3, 00:04:34, GigabitEthernet5/0
O IA 192.168.3.0/24 [110/3] via 133.10.2.3, 00:03:57, GigabitEthernet5/0

O IA 192.168.4.0/24 [110/4] via 133.10.2.3, 00:03:57, GigabitEthernet5/0
O IA 192.168.5.0/24 [110/3] via 133.10.2.3, 00:04:34, GigabitEthernet5/0
O IA 192.168.6.0/24 [110/3] via 133.10.2.3, 00:03:57, GigabitEthernet5/0
C    192.168.7.0/24 is directly connected, FastEthernet0/0
O IA 192.168.8.0/24 [110/3] via 133.10.2.3, 00:03:57, GigabitEthernet5/0
O IA 192.168.9.0/24 [110/3] via 133.10.2.3, 00:03:57, GigabitEthernet5/0
O IA 192.168.10.0/24 [110/5] via 133.10.2.3, 00:03:57, GigabitEthernet5/0
O IA 192.168.12.0/24 [110/4] via 133.10.2.3, 00:04:34, GigabitEthernet5/0
O IA 192.168.13.0/24 [110/4] via 45.10.2.3, 00:03:57, GigabitEthernet4/0
O IA 192.168.14.0/24 [110/5] via 133.10.2.3, 00:03:57, GigabitEthernet5/0
O IA 192.168.15.0/24 [110/4] via 133.10.2.3, 00:03:57, GigabitEthernet5/0
O IA 192.168.17.0/24 [110/5] via 133.10.2.3, 00:04:34, GigabitEthernet5/0
```

Figura 106. Tabla de enrutamiento de Suiza, parte 1.

```
O IA 192.168.18.0/24 [110/6] via 133.10.2.3, 00:04:34, GigabitEthernet5/0
O IA 192.168.19.0/24 [110/6] via 133.10.2.3, 00:04:34, GigabitEthernet5/0
O IA 192.168.21.0/24 [110/6] via 133.10.2.3, 00:04:34, GigabitEthernet5/0
O IA 192.168.22.0/24 [110/6] via 133.10.2.3, 00:04:34, GigabitEthernet5/0
O IA 192.168.24.0/24 [110/5] via 133.10.2.3, 00:03:57, GigabitEthernet5/0
O    192.168.31.0/24 [110/2] via 47.10.2.4, 00:04:34, GigabitEthernet1/0
O    192.168.32.0/24 [110/3] via 47.10.2.4, 00:04:34, GigabitEthernet1/0
O IA 192.168.33.0/24 [110/6] via 133.10.2.3, 00:03:57, GigabitEthernet5/0
O    192.168.35.0/24 [110/4] via 45.10.2.3, 00:04:34, GigabitEthernet4/0
O    192.168.36.0/24 [110/3] via 45.10.2.3, 00:04:34, GigabitEthernet4/0
O    192.168.37.0/24 [110/2] via 44.10.2.3, 00:04:34, GigabitEthernet7/0
O    192.168.38.0/24 [110/4] via 44.10.2.3, 00:04:34, GigabitEthernet7/0
                     [110/4] via 45.10.2.3, 00:04:34, GigabitEthernet4/0
O    192.168.39.0/24 [110/4] via 45.10.2.3, 00:04:34, GigabitEthernet4/0
                     [110/4] via 133.10.2.3, 00:03:57, GigabitEthernet5/0
O    192.168.40.0/24 [110/4] via 45.10.2.3, 00:04:34, GigabitEthernet4/0
O    192.168.43.0/24 [110/3] via 44.10.2.3, 00:04:34, GigabitEthernet7/0
                     [110/3] via 45.10.2.3, 00:04:34, GigabitEthernet4/0
O    192.168.44.0/24 [110/2] via 45.10.2.3, 00:04:34, GigabitEthernet4/0
O    192.168.46.0/24 [110/3] via 47.10.2.4, 00:04:34, GigabitEthernet1/0
O IA 192.168.53.0/24 [110/6] via 133.10.2.3, 00:03:57, GigabitEthernet5/0
O IA 192.168.67.0/24 [110/7] via 133.10.2.3, 00:04:34, GigabitEthernet5/0
O IA 192.168.73.0/24 [110/7] via 133.10.2.3, 00:03:57, GigabitEthernet5/0
O IA 192.168.74.0/24 [110/4] via 133.10.2.3, 00:04:34, GigabitEthernet5/0
O IA 192.168.75.0/24 [110/4] via 133.10.2.3, 00:04:34, GigabitEthernet5/0
O IA 192.168.77.0/24 [110/6] via 133.10.2.3, 00:03:57, GigabitEthernet5/0
O IA 192.168.78.0/24 [110/5] via 133.10.2.3, 00:03:57, GigabitEthernet5/0
O IA 192.168.79.0/24 [110/6] via 133.10.2.3, 00:03:57, GigabitEthernet5/0
O IA 192.168.80.0/24 [110/7] via 133.10.2.3, 00:03:57, GigabitEthernet5/0
O    193.10.2.0/24 [110/3] via 45.10.2.3, 00:04:34, GigabitEthernet4/0

O    194.10.2.0/24 [110/2] via 45.10.2.3, 00:04:34, GigabitEthernet4/0
O    195.10.2.0/24 [110/3] via 45.10.2.3, 00:04:34, GigabitEthernet4/0
O    196.10.2.0/24 [110/3] via 45.10.2.3, 00:04:34, GigabitEthernet4/0
O IA 197.10.2.0/24 [110/3] via 45.10.2.3, 00:04:34, GigabitEthernet4/0
O IA 198.10.2.0/24 [110/3] via 45.10.2.3, 00:04:34, GigabitEthernet4/0
                   [110/3] via 133.10.2.3, 00:03:57, GigabitEthernet5/0
O    199.10.2.0/24 [110/3] via 45.10.2.3, 00:04:34, GigabitEthernet4/0
```

Figura 106.1. Tabla de enrutamiento de Suiza, parte 2.

En comparación mostramos la tabla de enrutamiento de Noruega y comprobamos que ambas tablas tienen las mismas redes, sin embargo, observamos que las redes “IA” son diferentes dado que ambos routers están conectados con diferentes áreas como se muestra en las figuras 107-107.1.

```
O IA 10.0.0.0/8 [110/4] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
O IA 12.0.0.0/8 [110/4] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
O IA 14.0.0.0/8 [110/4] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
O IA 15.0.0.0/8 [110/4] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
O IA 16.0.0.0/8 [110/4] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
O IA 17.0.0.0/8 [110/4] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
O IA 18.0.0.0/8 [110/4] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
O    22.0.0.0/8 [110/2] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
O IA 40.0.0.0/8 [110/5] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
O IA 41.0.0.0/8 [110/5] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
O IA 42.0.0.0/8 [110/5] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
O IA 43.0.0.0/8 [110/6] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
O IA 44.0.0.0/8 [110/5] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
O IA 45.0.0.0/8 [110/5] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
O IA 46.0.0.0/8 [110/5] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
O IA 47.0.0.0/8 [110/5] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
O IA 48.0.0.0/8 [110/6] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
O IA 49.0.0.0/8 [110/6] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
O IA 51.0.0.0/8 [110/4] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
O IA 52.0.0.0/8 [110/5] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
O IA 53.0.0.0/8 [110/5] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
O IA 54.0.0.0/8 [110/5] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
O IA 90.0.0.0/8 [110/5] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
O IA 91.0.0.0/8 [110/5] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
O IA 92.0.0.0/8 [110/5] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
O    93.0.0.0/8 [110/5] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
```

Figura 107. Tabla de enrutamiento de Noruega, parte 1.

```
O    94.0.0.0/8 [110/4] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
C    95.0.0.0/8 is directly connected, GigabitEthernet1/0
C    96.0.0.0/8 is directly connected, GigabitEthernet2/0
O    98.0.0.0/8 [110/2] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
O IA 99.0.0.0/8 [110/3] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
O    100.0.0.0/8 [110/3] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
O    101.0.0.0/8 [110/2] via 96.10.2.4, 00:16:09, GigabitEthernet2/0
O    129.10.0.0/16 [110/4] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
O    130.10.0.0/16 [110/3] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
O IA 131.10.0.0/16 [110/4] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
O    132.10.0.0/16 [110/4] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
O IA 133.10.0.0/16 [110/4] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
O    134.10.0.0/16 [110/4] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
O    135.10.0.0/16 [110/4] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
O IA 136.10.0.0/16 [110/3] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
O IA 137.10.0.0/16 [110/3] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
O IA 138.10.0.0/16 [110/3] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
O    139.10.0.0/16 [110/3] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
O    140.10.0.0/16 [110/3] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
O    141.10.0.0/16 [110/3] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
O    142.10.0.0/16 [110/2] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
O IA 150.10.0.0/16 [110/4] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
O IA 151.10.0.0/16 [110/5] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
O IA 152.10.0.0/16 [110/5] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
O IA 153.10.0.0/16 [110/5] via 95.10.2.3, 00:16:09, GigabitEthernet1/0

O IA 154.10.0.0/16 [110/4] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
O IA 155.10.0.0/16 [110/5] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
O IA 156.10.0.0/16 [110/5] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
O IA 157.10.0.0/16 [110/4] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
O IA 158.10.0.0/16 [110/4] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
O IA 159.10.0.0/16 [110/4] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
O IA 192.10.2.0/24 [110/4] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
O    192.168.2.0/24 [110/4] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
O    192.168.3.0/24 [110/3] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
O    192.168.4.0/24 [110/3] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
O IA 192.168.5.0/24 [110/5] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
O    192.168.6.0/24 [110/4] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
O IA 192.168.7.0/24 [110/5] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
O    192.168.8.0/24 [110/5] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
O    192.168.9.0/24 [110/5] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
O IA 192.168.10.0/24 [110/4] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
O IA 192.168.12.0/24 [110/4] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
O    192.168.13.0/24 [110/4] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
O    192.168.14.0/24 [110/4] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
O    192.168.15.0/24 [110/2] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
O IA 192.168.17.0/24 [110/5] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
O IA 192.168.18.0/24 [110/6] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
O IA 192.168.19.0/24 [110/5] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
O IA 192.168.21.0/24 [110/6] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
O IA 192.168.22.0/24 [110/5] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
O IA 192.168.24.0/24 [110/5] via 95.10.2.3, 00:16:09, GigabitEthernet1/0

O IA 192.168.31.0/24 [110/6] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
O IA 192.168.32.0/24 [110/7] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
O IA 192.168.33.0/24 [110/5] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
O IA 192.168.35.0/24 [110/5] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
O IA 192.168.36.0/24 [110/4] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
O IA 192.168.37.0/24 [110/6] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
O IA 192.168.38.0/24 [110/5] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
O IA 192.168.39.0/24 [110/4] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
O IA 192.168.40.0/24 [110/3] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
O IA 192.168.43.0/24 [110/6] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
O IA 192.168.44.0/24 [110/5] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
O IA 192.168.46.0/24 [110/7] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
O IA 192.168.53.0/24 [110/6] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
O IA 192.168.67.0/24 [110/6] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
O    192.168.73.0/24 [110/6] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
O IA 192.168.74.0/24 [110/6] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
O IA 192.168.75.0/24 [110/6] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
O    192.168.77.0/24 [110/5] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
C    192.168.78.0/24 is directly connected, FastEthernet0/0
O    192.168.79.0/24 [110/2] via 96.10.2.4, 00:16:09, GigabitEthernet2/0
O    192.168.80.0/24 [110/3] via 96.10.2.4, 00:16:09, GigabitEthernet2/0
O IA 193.10.2.0/24 [110/4] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
O IA 194.10.2.0/24 [110/4] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
O IA 195.10.2.0/24 [110/4] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
O IA 196.10.2.0/24 [110/4] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
O    197.10.2.0/24 [110/4] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
O    198.10.2.0/24 [110/5] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
O IA 199.10.2.0/24 [110/3] via 95.10.2.3, 00:16:09, GigabitEthernet1/0
```

Figura 107.1. Tabla de enrutamiento de Noruega, parte 2.

5.1.11. Costos OSPF en la simulación de la red GEANT

En esta sección comprobamos la métrica de OSPF, la cual funciona eligiendo la ruta con el menor costo y como ejemplo realizaremos la prueba entre los host de Irlanda y Reino Unido, los cuales se conectan con 2 interfaces cuyos anchos de banda son de 10 Mb y 1 GB, dando como resultado un costo de 100 y 1 como se muestra en la figura 108.

```
Router#sh ip ospf int

GigabitEthernet1/0 is up, line protocol is up
  Internet address is 192.10.2.4/24, Area 1
  Process ID 1, Router ID 193.10.2.4, Network Type BROADCAST, Cost: 100
  Transmit Delay is 1 sec, State BDR, Priority 1
  Designated Router (ID) 199.10.2.3, Interface address 192.10.2.1
  Backup Designated Router (ID) 193.10.2.4, Interface address 192.10.2.4
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    Hello due in 00:00:09
  Index 1/1, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 1, maximum is 1
  Last flood scan time is 0 msec, maximum is 0 msec
  Neighbor Count is 1, Adjacent neighbor count is 1
    Adjacent with neighbor 199.10.2.3  (Designated Router)
  Suppress hello for 0 neighbor(s)
GigabitEthernet2/0 is up, line protocol is up
  Internet address is 193.10.2.4/24, Area 1
  Process ID 1, Router ID 193.10.2.4, Network Type BROADCAST, Cost: 1
  Transmit Delay is 1 sec, State BDR, Priority 1
  Designated Router (ID) 199.10.2.3, Interface address 193.10.2.3
  Backup Designated Router (ID) 193.10.2.4, Interface address 193.10.2.4
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    Hello due in 00:00:09
  Index 2/2, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 1, maximum is 1
  Last flood scan time is 0 msec, maximum is 0 msec
  Neighbor Count is 1, Adjacent neighbor count is 1
    Adjacent with neighbor 199.10.2.3  (Designated Router)
  Suppress hello for 0 neighbor(s)
```

Figura 108. Configuración de OSPF en router de Irlanda.

Comprobamos que el algoritmo de OSPF elige la interfaz G2/0 con la dirección de red 193.10.2.3 con un costo de 1 como se muestra en la figuras 109 y 110.

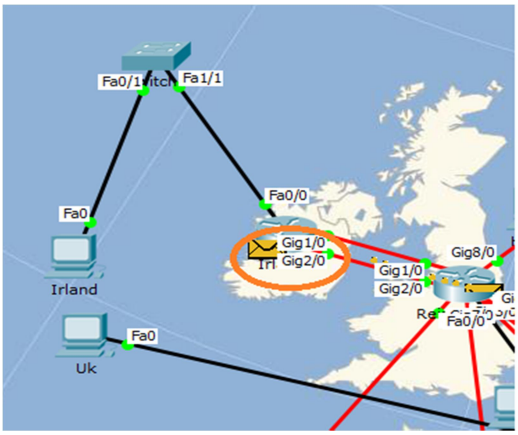


Figura 109. Prueba de conectividad de Irlanda a UK.

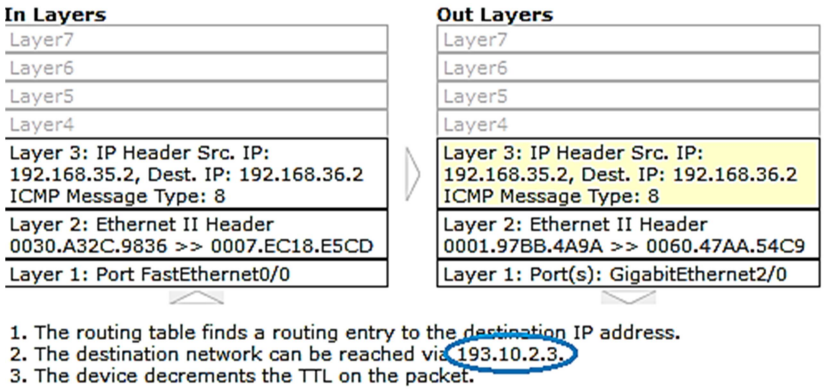


Figura 110. Demostración de interfaz G2/0.

5.1.12. Paquete Hello de OSPF en simulación de GEANT

El simulador Packet Tracer nos ofrece de forma gráfica el contenido del paquete "*Hello*" del protocolo OSPF v2, como lo vimos anteriormente su encabezado cuenta con el campo de versión del protocolo, el tipo de mensaje con un valor de "1" que en este caso identifica al paquete "*Hello*" como se muestra en la figura 111.

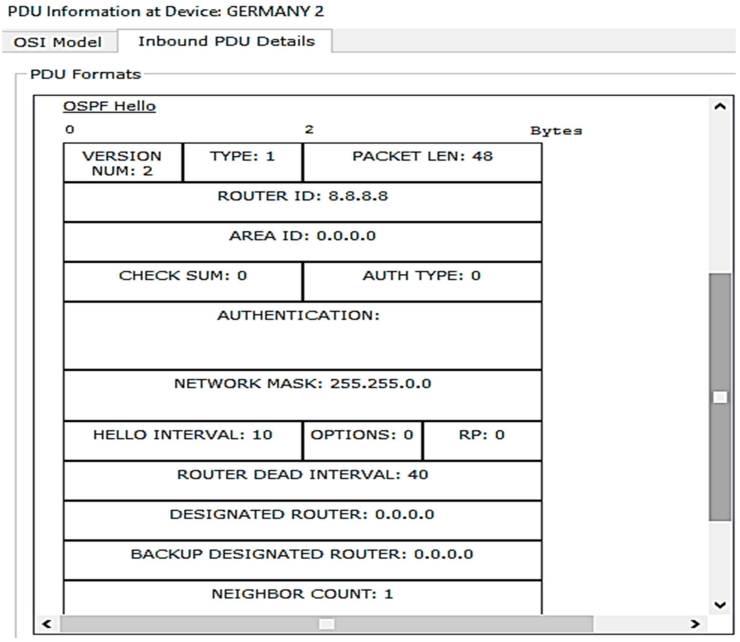


Figura 111. Paquete *Hello* del Router Alemania 2

Este mensaje también despliega información del ROUTER-ID: 8.8.8.8 que definimos previamente en el protocolo OSPF para el router de Alemania 2, el ÁREA-ID: 0.0.0.0 que nos indica el tipo de área, el

campo de autentificación y el campo "*Check sum*" para la de detección de errores con la suma binaria del paquete IP. Visto lo anterior, el formato de mensaje *Hello* nos muestra información de los valores condicionales para crear adyacencia entre los routers visto en el capítulo 3 (Sección 3.4.7).

5.1.13. Resilencia y redundancia en la simulación de GEANT con OSPF v2

En esta sección pondremos a prueba el protocolo OSPF v2 al mandar paquetes ICMP, simultáneos desde distintos nodos de la red y deshabilitando el router de Alemania para comprobar la latencia y la velocidad de convergencia ante un evento no esperado como lo hicimos previamente con el protocolo RIP v2. La prueba será exactamente la misma que se realizó anteriormente con el protocolo RIP v2 entre los nodos de UK-Bélgica, Alemania-España, Suiza-Turquía, Francia 1-Ucrania, Austria-Irlanda, Hungria-Letonia y Noruega-Estonia como se muestran en la figuras 112-118.

```
Reply from 192.168.39.2: bytes=32 time=0ms TTL=126
Reply from 192.168.39.2: bytes=32 time=0ms TTL=126
Reply from 192.168.39.2: bytes=32 time=0ms TTL=126
Reply from 192.168.39.2: bytes=32 time=1ms TTL=126
Reply from 192.168.39.2: bytes=32 time=0ms TTL=126
Reply from 192.168.39.2: bytes=32 time=0ms TTL=126
Reply from 192.168.39.2: bytes=32 time=1ms TTL=126
```

Figura 112. Prueba de conectividad simultánea UK-Bélgica.

```
Reply from 192.168.2.1: Destiantion host unreachable
Request timed out.
Reply from 192.168.2.1: Destiantion host unreachable
Request timed out.
Reply from 192.168.2.1: Destiantion host unreachable
Request timed out.
Reply from 192.168.2.1: Destiantion host unreachable
Reply from 192.168.2.1: Destiantion host unreachable
```

Figura 113. Prueba de conectividad simultánea Alemania-España.

```
Reply from 192.168.2.1: Destiantion host unreachable
Reply from 192.168.2.1: Destiantion host unreachable
Reply from 192.168.2.1: Destiantion host unreachable
Request timed out.
Reply from 192.168.2.1: Destiantion host unreachable
Reply from 192.168.2.1: Destiantion host unreachable
```

Figura 114. Prueba de conectividad simultánea Suiza-Turquía.

```
Reply from 192.168.75.2: bytes=32 time=12ms TTL=120
Reply from 192.168.75.2: bytes=32 time=12ms TTL=120
Reply from 192.168.75.2: bytes=32 time=11ms TTL=120
Reply from 192.168.75.2: bytes=32 time=11ms TTL=120
Reply from 192.168.75.2: bytes=32 time=10ms TTL=120
Reply from 192.168.75.2: bytes=32 time=11ms TTL=120
Reply from 192.168.75.2: bytes=32 time=11ms TTL=120
```

Figura 115. Prueba de conectividad simultánea Francia 1-Ucrania.

```
Reply from 192.168.35.2: bytes=32 time=11ms TTL=123
Reply from 192.168.35.2: bytes=32 time=0ms TTL=123
Reply from 192.168.35.2: bytes=32 time=0ms TTL=123
Reply from 192.168.35.2: bytes=32 time=1ms TTL=123
Reply from 192.168.35.2: bytes=32 time=1ms TTL=123
Reply from 192.168.35.2: bytes=32 time=0ms TTL=123
Reply from 192.168.35.2: bytes=32 time=0ms TTL=123
```

Figura 116. Prueba de conectividad simultánea Austria-Irlanda.

```
Reply from 192.168.77.2: bytes=32 time=1ms TTL=123
Reply from 192.168.77.2: bytes=32 time=10ms TTL=123
Reply from 192.168.77.2: bytes=32 time=11ms TTL=123
Reply from 192.168.77.2: bytes=32 time=0ms TTL=123
Reply from 192.168.77.2: bytes=32 time=0ms TTL=123
Reply from 192.168.77.2: bytes=32 time=12ms TTL=123
```

Figura 117. Prueba de conectividad simultánea Hungria-Letonia.

```
Reply from 192.168.14.2: bytes=32 time=0ms TTL=124
Reply from 192.168.14.2: bytes=32 time=2ms TTL=124
Reply from 192.168.14.2: bytes=32 time=0ms TTL=124
Reply from 192.168.14.2: bytes=32 time=15ms TTL=124
Reply from 192.168.14.2: bytes=32 time=0ms TTL=124
Reply from 192.168.14.2: bytes=32 time=11ms TTL=124
```

Figura 118. Prueba de conectividad simultánea Noruega-Estonia.

En las figuras anteriores observamos que la comunicación en los nodos de Alemania 2 y Suiza quedaron sin conectividad, esto se debió a que el router de Alemania fue deshabilitado por completo, mientras que el nodo de Suiza, dependía por completo del router de Alemania ya que, este es un router ABR que conecta el área 0 y el área 2. Por otro lado el resto de los nodos no tuvieron ninguna afectación, respecto a la latencia los paquetes sufrieron un retardo de entre 1 y 11 milisegundos por prueba.

5.2. Gestión de GEANT con SNMP en simulación

Una vez configurada la MIB browser del host de Alemania2, podemos realizar operaciones de tipo *Get* o *Set* en el router por medio del árbol MIB del simulador, agregando la OID del objeto en la consola de MIB o desplazando el árbol MIB hasta el objeto deseado. Es decir si nosotros queremos obtener información de las rutas de destino del router Alemania2 sin acceder a la consola del router,

desplegamos el árbol MIB hasta la OID que representa las rutas IP destino y elegimos la operación de tipo *Get* como se muestra en la figura 119.

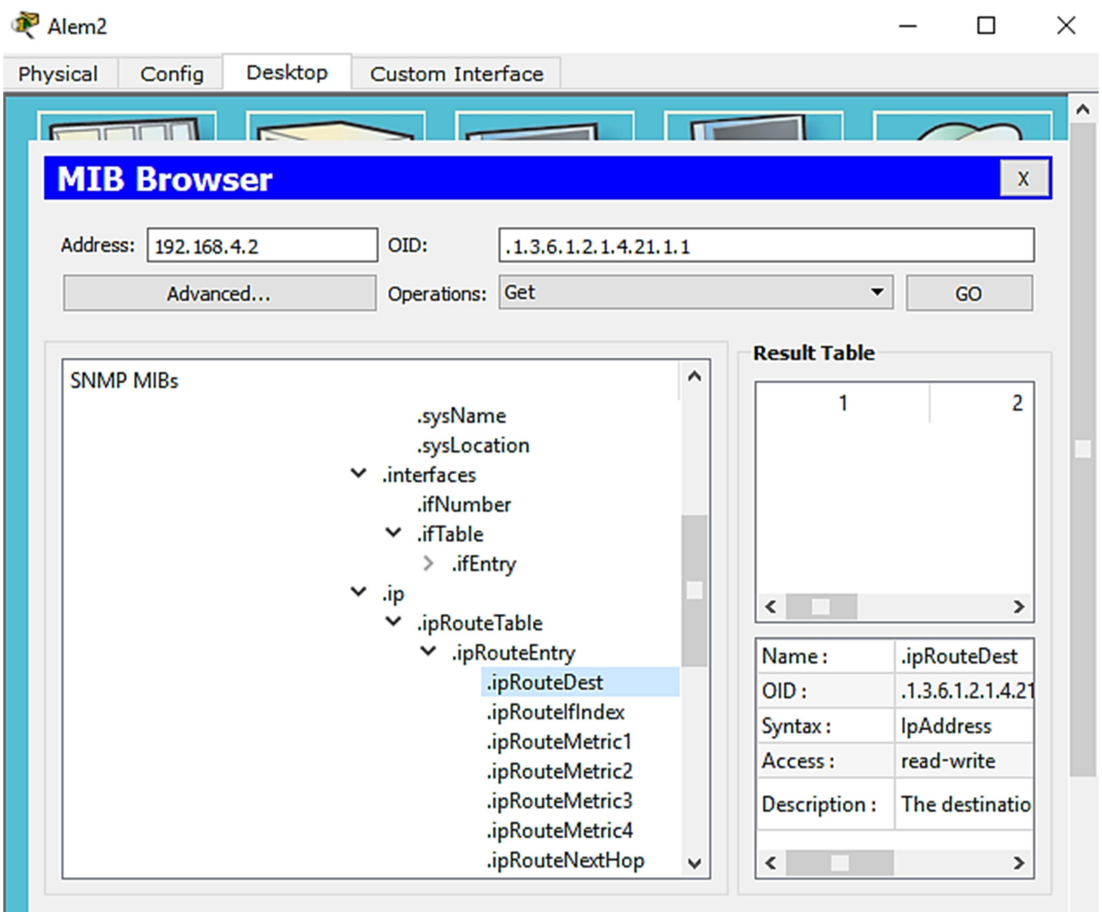


Figura 119. MIB Tree en simulación.

5.2.1. *SNMP Object Navigator* Cisco

El árbol MIB del simulador Packet Tracer nos despliega la OID 1.3.6.1.2.1.4.21.1.1 que representa las direcciones IP destino del router de Alemania2. Para corroborar que efectivamente la OID 1.3.6.1.2.1.4.21.1.1 pertenece al objeto de las direcciones IP destino, usaremos una herramienta que nos proporciona cisco en su página *Web*. Para esto nos dirigimos a la página *Web* de Cisco e ingresamos en la herramienta *SNMP Objet Navigator*. Seleccionamos la opción ***Cisco Tools & Resources → SNMP Object Navigator*** como se muestra en la figura 120.

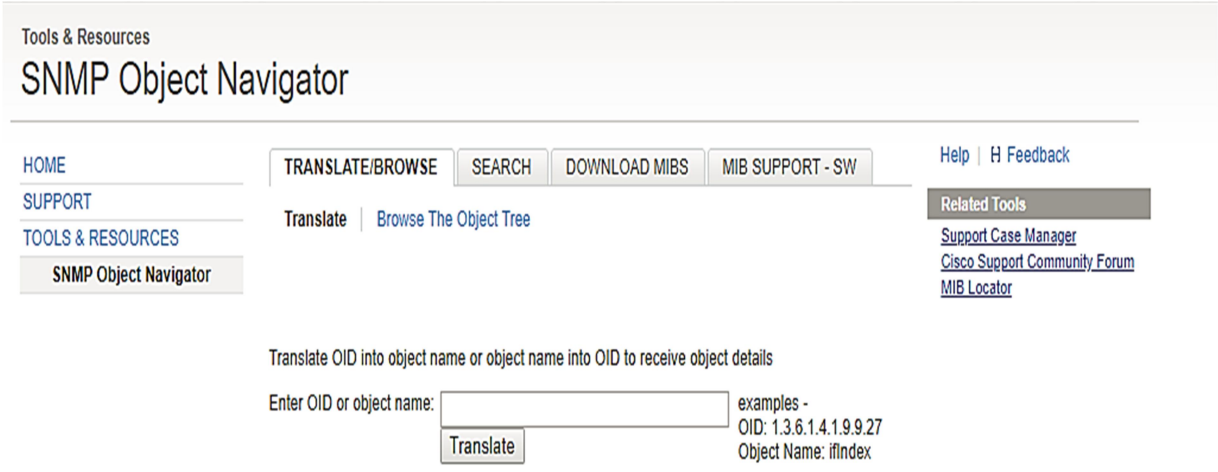


Figura 120. Herramienta SNMP Object Navigator (Cisco Web

Después ingresamos la OID que nos proporcionó el árbol MIB del simulador en el campo "*Enter OID or object name"* como se muestra en la figura 121.

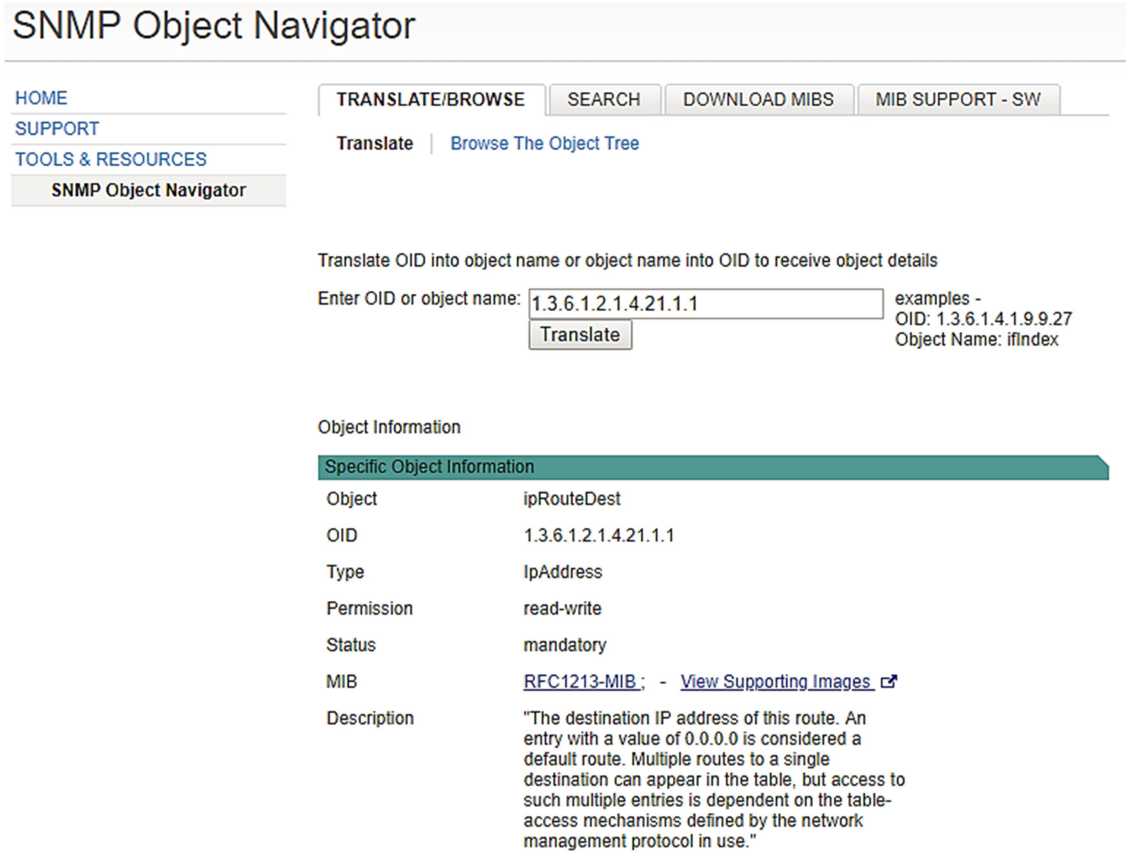


Figura 121. Cisco SNMP Object.

El resultado de la herramienta de cisco con la OID 1.3.6.1.2.1.4.21.1.1, especifica las rutas IP destino de la tabla de enrutamiento.

5.2.2. Operación *Set*-SNMP en simulación de GEANT

Los mensajes de tipo *Get* nos permitirán gestionar y modificar ciertos valores del router a distancia. Como prueba de esto seleccionaremos el objeto con la OID .1.3.6.1.2.1.1.5.0 en el árbol MIB que representa el nombre del router como se muestra en la figura 122.

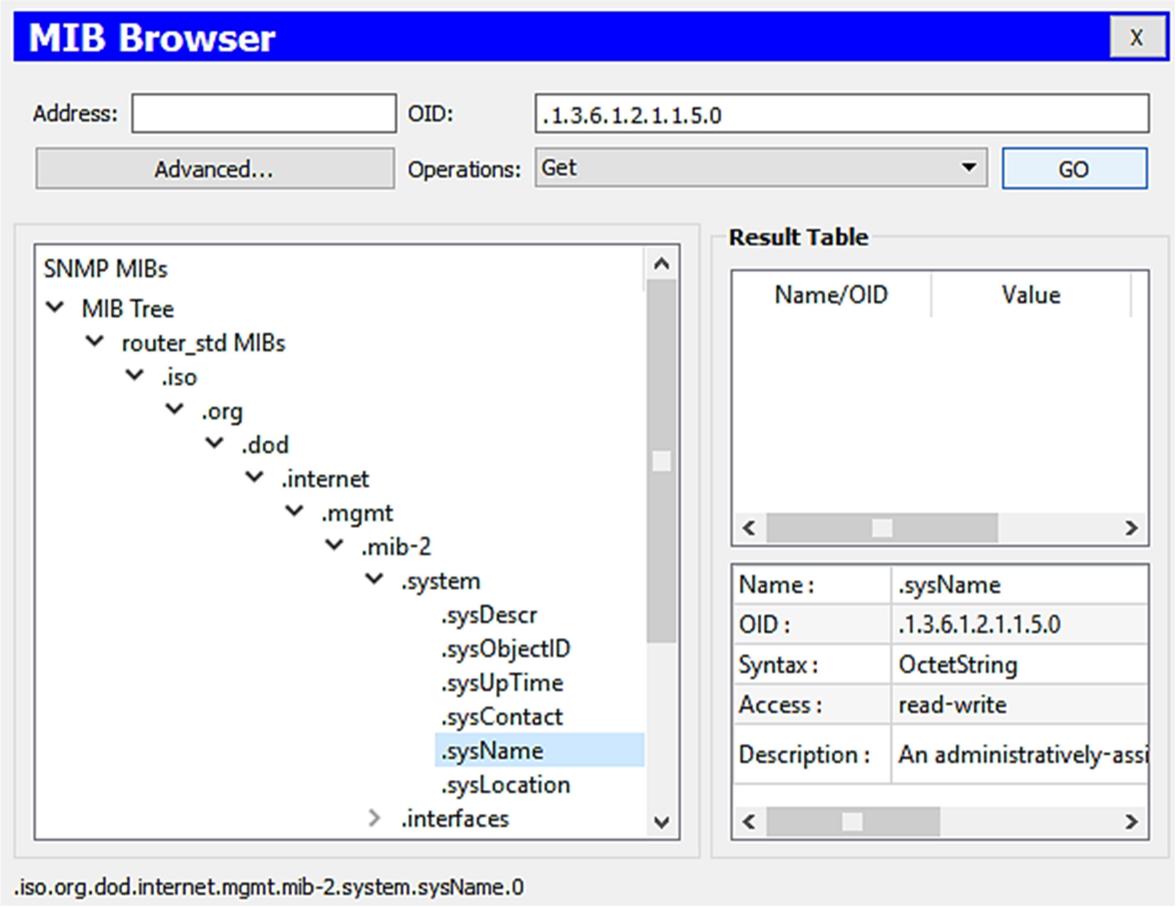


Figura 122. Árbol MIB Sys Name

Como mencionamos anteriormente la OID identifica el valor del nombre del router y para poder cambiar el valor del nombre del router elegimos la opción "*Set*" en el MIB Browser del simulador, elegimos el tipo de dato a cambiar "*Data Type"*, seleccionamos el valor Octetstring y agregamos el nuevo valor en el campo *Value,* es decir, el nuevo nombre del router como se muestra en la figura 123.

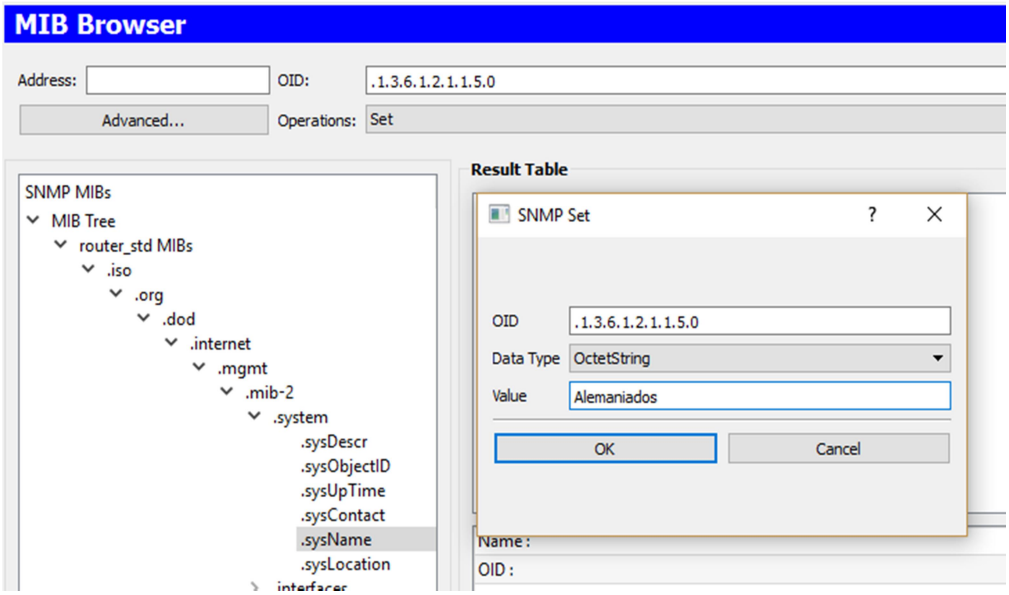


Figura 123. SNMP Set.

Después de haber llenado los campos correctamente y aceptar los cambios el mensaje de tipo *Set-SNM*P habrá cambiado el valor del nombre del router a "Alemaniados" como se muestran en las figuras 124 y 125.

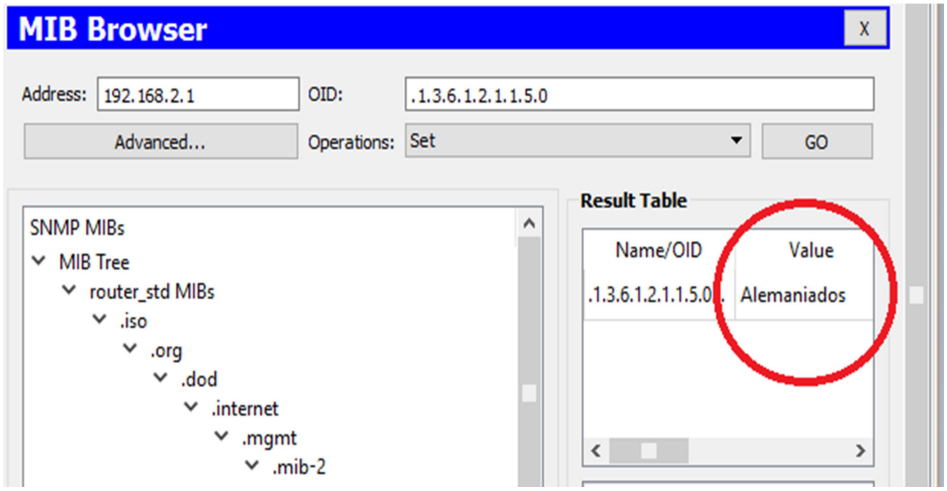


Figura 124. MIB browser Sys Name Router alemania2.

Figura 125. Consola de Router de Alemania2.

5.3.1. Estados OSPF en la emulación de GEANT

A continuación mostraremos el proceso de adyacencia entre los routers después de configurar OSPF v2 en la emulación. El software Wireshark nos ofrece una plataforma grafica que muestra de manera general el proceso de convergencia OSPF y despliega información de cada paquete. Como ejemplo analizaremos la conectividad entre Finlandia y Escocia como se muestra en la figura 126.

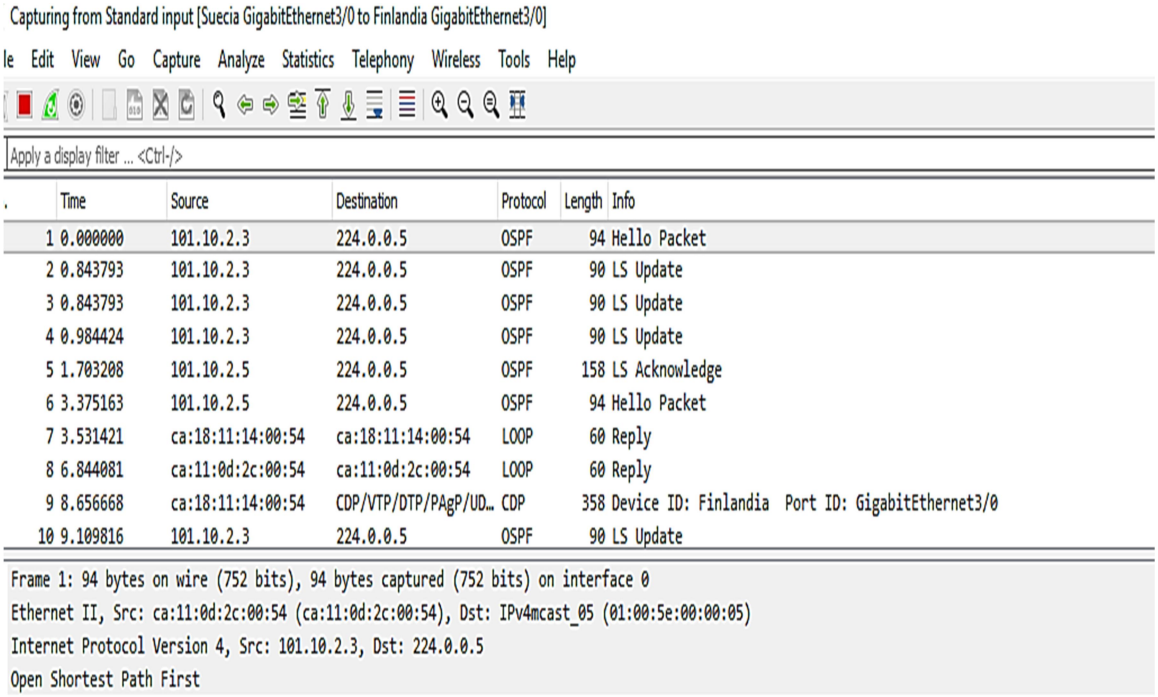


Figura 126. Captura de paquetes OSPF en Wireshark

Wireshark nos despliega informacion de los mensajes que se envian en cada conexion con información del tiempo en que se generan, la fuente (source), es decir, en donde se originaron, la multidifision, el tipo de protocolo, la longitud del paquete (length) y por ultimo el tipo de mensaje que se esta enviando. Dicho esto en la figura 176 podemos verificar el proceso de adyacencia con los paquetes *Hello, LSA Update y LSACK* de paquetes OSPF que se envián y se repiten hasta llegar al estado de convergencia OSPF.

5.3.2. Cabecera del paquete OSPF

Analizando los mensajes OSPF en Wireshark, este nos indica el formato de cada paquete. Por ejemplo la cabecera del mensaje OSPF nos describe el tipo de versión del protocolo, el tipo de mensaje que es.

En este caso analizaremos el paquete LSU, visto en la sección 3.4.11 del capítulo 3, el área del router que pertenece al área 1 como se muestra en la figura 127.

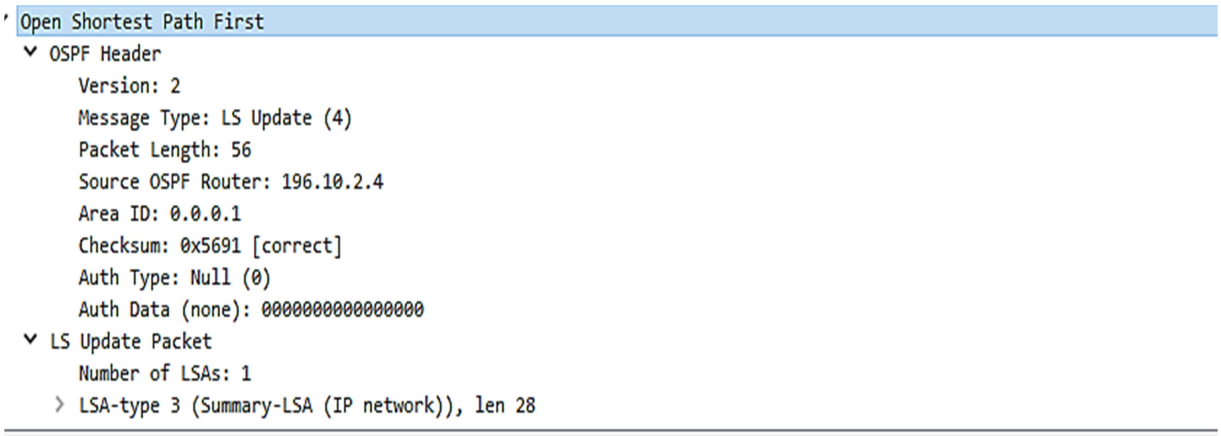


Figura 127. Cabecera de mensaje OSPF en interface de Bélgica con Wireshark.

5.3.3. Paquete *Hello* OSPF en emulación

El paquete "*Hello*" en la emulación nos proporciona información que nos permite analizar los criterios de adyacencia de OSPF v2, como la máscara de la interfaz de red, el intervalo del paquete, la prioridad del router, el tiempo del intervalo muerto y el tipo de router en este caso es un ABR como se muestra en la figura 128.

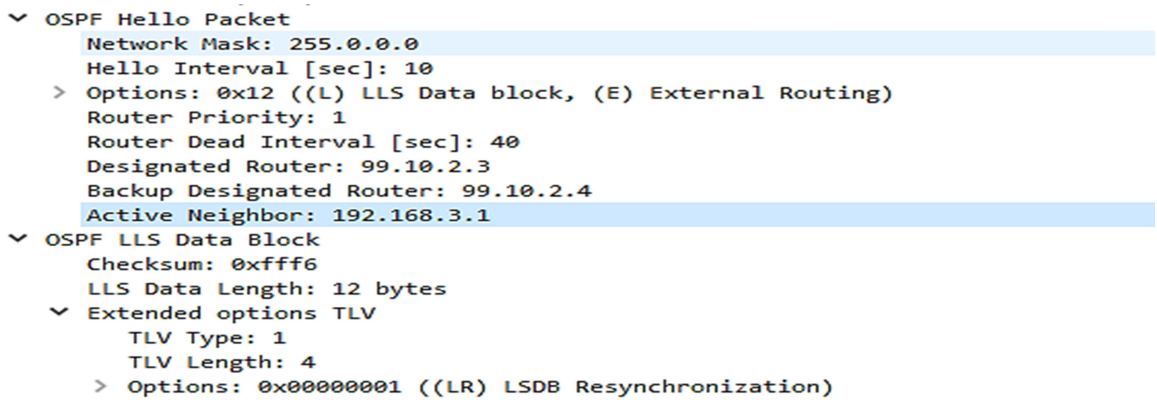


Figura 128. Paquete Hello del router de Holanda en emulación.

5.3.5. Prueba de conectividad en la emulación de GEANT

En esta sección comprobaremos la conectividad en nuestra emulación de red, por medio del comando ping. En comparación con la simulación, el emulador nos ofrece únicamente el modo CLI por medio

de las consolas de los Host, routers o máquinas virtuales. Comenzaremos con la prueba por medio de la máquina virtual de Alemania hacia todos los PoP de la red, como se muestra a continuación. Ver figura 129 y 130.

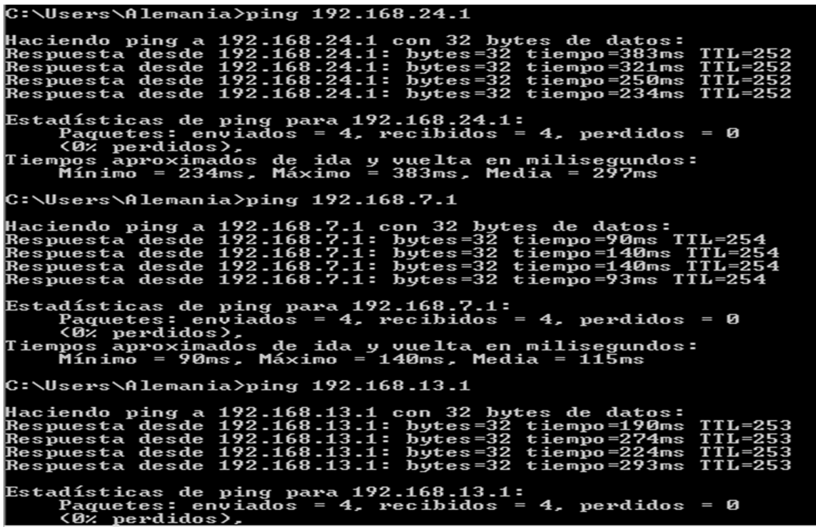


Figura 129. Prueba de conectividad para Bulgaria, Suiza y Chipre.

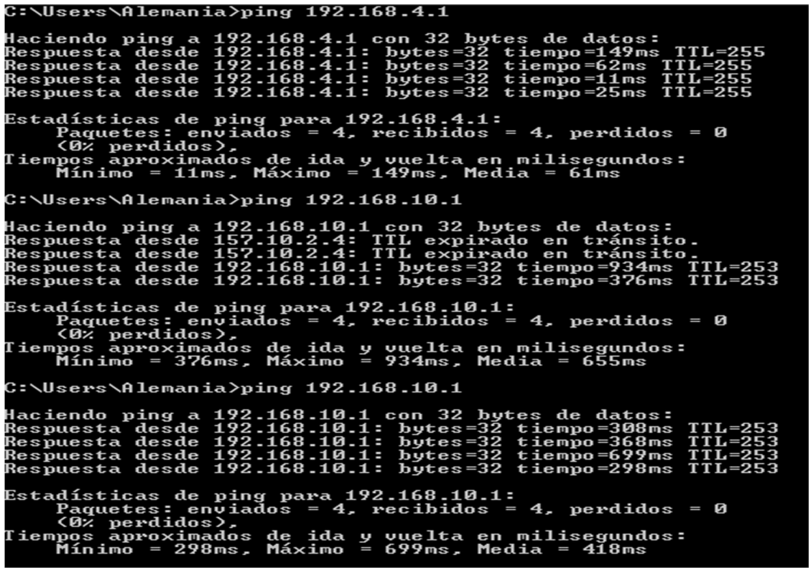


Figura 130. Prueba de conectividad de Alemania 2 para Alemania y Austria.

5.3.6. Prueba de conectividad por un Host (VPCS) en la emulación de GEANT

La prueba de conectividad por medio de un *Host* del emulador, se aplica de la misma manera que lo hicimos anteriormente. Elegimos el *Host* del PoP de suiza y realizamos Ping a los routers de Francia, Austria, Hungría, Polonia, España e Irlanda, como se muestra en la figura 131.

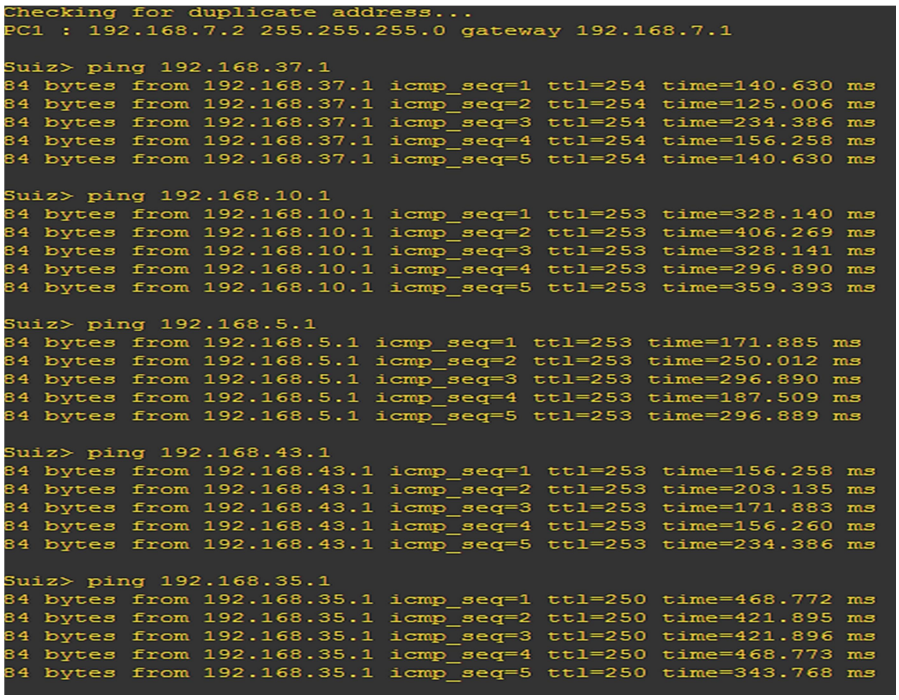


Figura 131. Prueba de conectividad a routers de GEANT en emulación.

5.3.7. Prueba de conectividad por medio de la consola del router en emulación

Por ultimo realizamos pruebas de conectividad con éxito desde el router de Holanda hacia los routers de Australia, Grecia, Islandia, noruega y Turquía como se muestra en la figura 132.

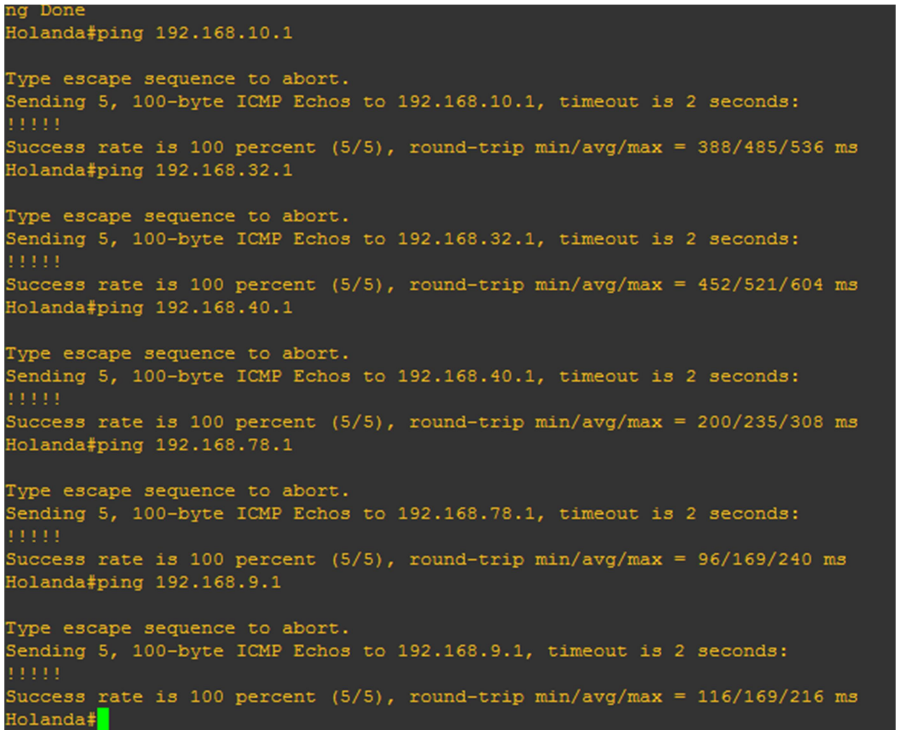


Figura 132. Prueba de conectividad del Router de Holanda hacia los routers de GEANT en emulación.

Hemos expuesto las tres maneras de hacer prueba de conectividad de manera satisfactoria en la emulación de GEANT, comprobando la correcta configuración de las interfaces de red y del protocolo OSPF v2 en toda la red. El resto de las pruebas se encuentran en el apéndice C.

5.3.8. ARP en Emulación de la red GEANT

Como mencionamos anteriormente la tabla ARP se utiliza como una relación entre las direcciones IP y las direcciones MAC de los routers, con el objetivo de crear adyacencias para generar un mapa de la red y generar una tabla de enrutamiento. Anteriormente en la simulación observamos que el proceso ARP consta de un conjunto de mensajes petición- respuesta, en donde se solicita por medio de la dirección IP, la dirección MAC del router vinculado a esa dirección. A continuación mostramos este proceso en la emulación de la red, al capturar información de un mensaje ARP entre el router de Portugal y Reino Unido. Ver figura 133 y 134.

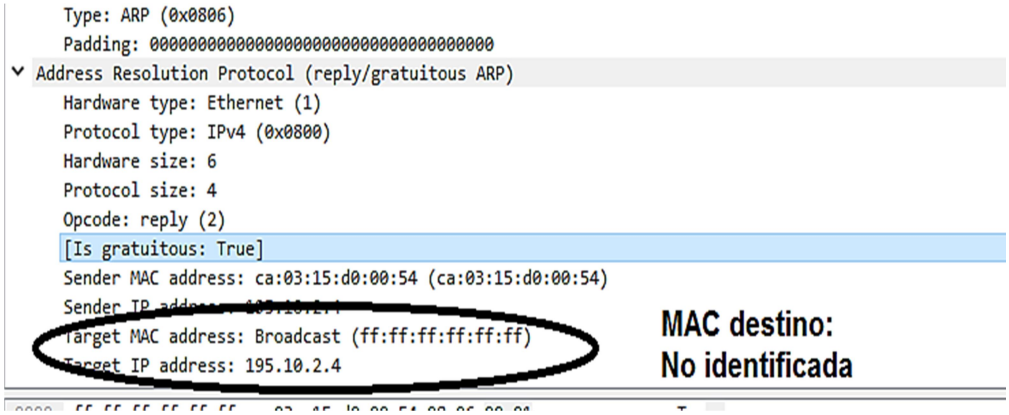


Figura 133. Dirección MAC no identificada del router de UK.

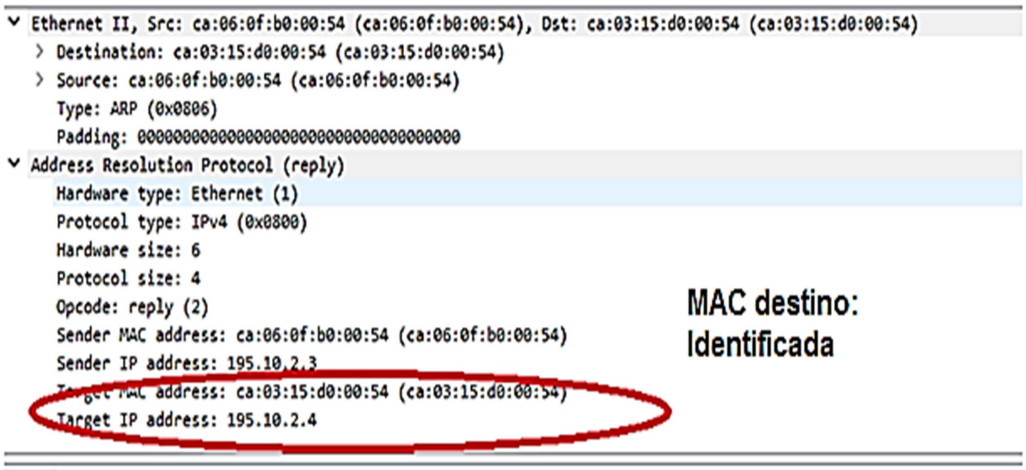


Figura 134. Dirección MAC identificada del router de UK.

Observamos que se logra el reconocimiento de la dirección física del router de UK creando las adyacencias correspondientes con el router solicitante para relacionar la dirección MAC con la dirección IP.

5.3.9. Prueba de Latencia con OSPF v2 en emulación de GEANT

En esta sección repetiremos la prueba que realizamos en la simulación para medir la residencia de la red ante la caída del router de Alemania. Los nodos con pruebas de conectividad simultáneas serán: UK-Bélgica, Alemania-España, Suiza-Turquía, Francia 1-Ucrania, Austria-Irlanda, Hungria-Letonia y Noruega-Estonia como se muestra en las figuras 135-141.

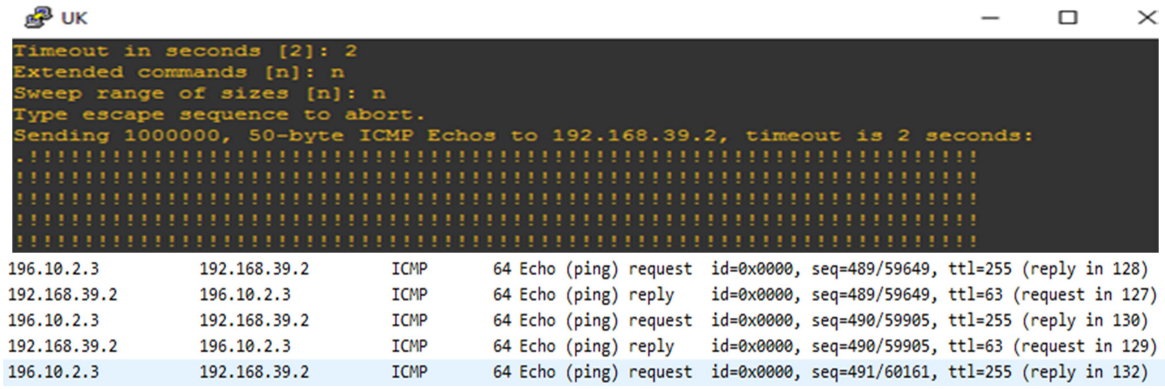


Figura 135. Prueba de conectividad simultánea en emulación UK- Bélgica.

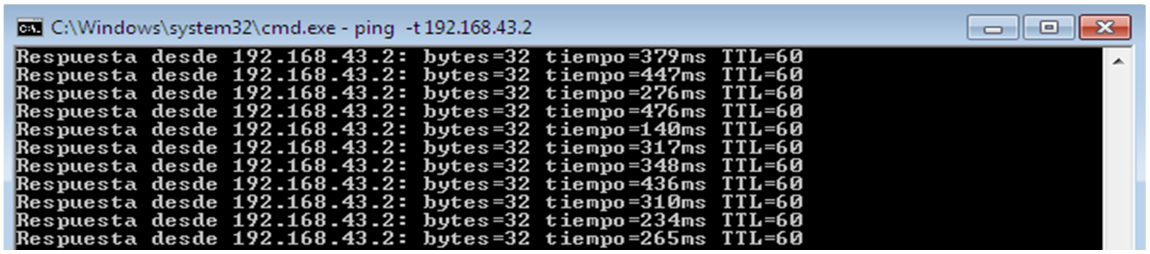
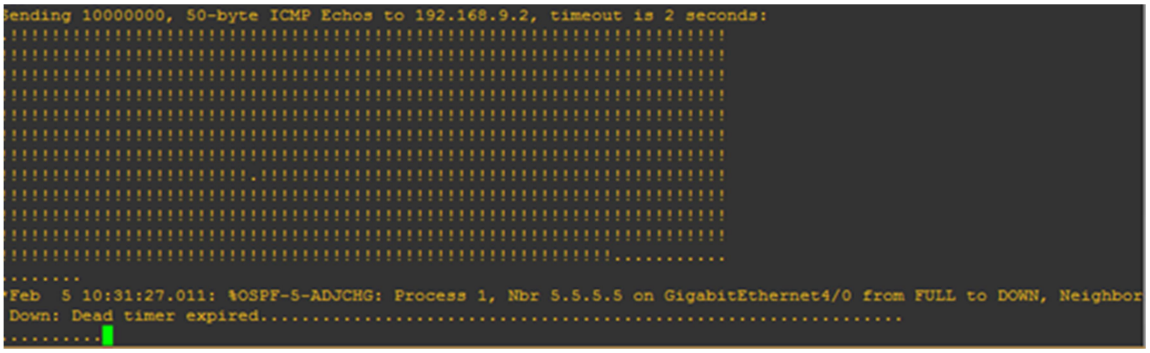
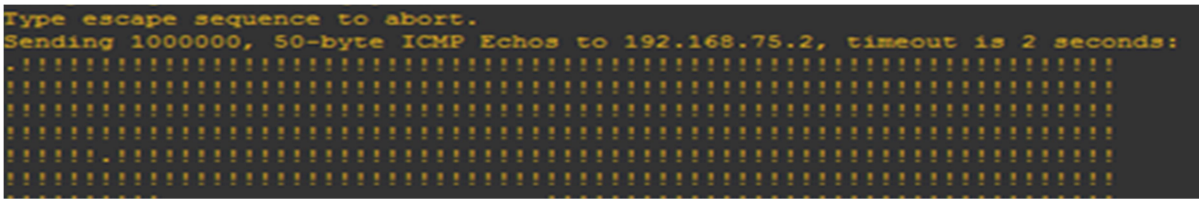


Figura 136. Prueba de conectividad simultánea en emulación Alemania-España.



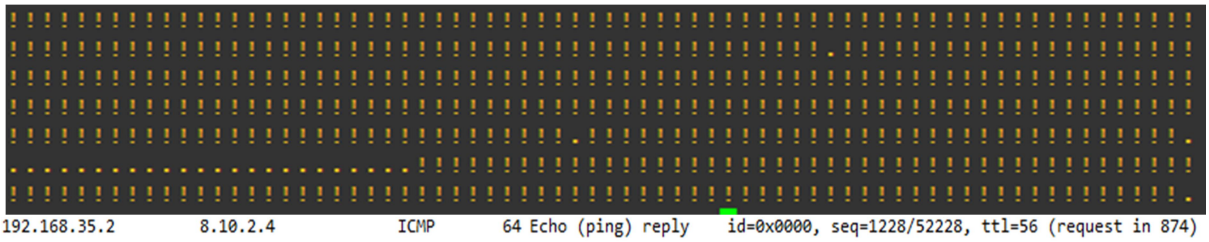
```
192.168.9.2     133.10.2.4      ICMP    64 Echo (ping) reply    id=0x0000, seq=68/17408, ttl=62 (request in 2)
133.10.2.4      192.168.9.2     ICMP    64 Echo (ping) request  id=0x0000, seq=69/17664, ttl=255 (reply in 5)
192.168.9.2     133.10.2.4      ICMP    64 Echo (ping) reply    id=0x0000, seq=69/17664, ttl=62 (request in 4)
133.10.2.4      192.168.9.2     ICMP    64 Echo (ping) request  id=0x0000, seq=70/17920, ttl=255 (reply in 7)
192.168.9.2     133.10.2.4      ICMP    64 Echo (ping) reply    id=0x0000, seq=70/17920, ttl=62 (request in 6)
133.10.2.4      192.168.9.2     ICMP    64 Echo (ping) request  id=0x0000, seq=71/18176, ttl=255 (reply in 9)
```

Figura 137. Prueba de conectividad simultánea en emulación Suiza Turquía.



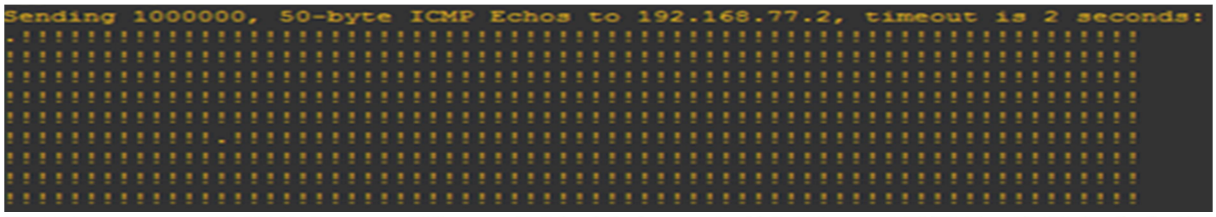

```
44.10.2.3       192.168.75.2    ICMP    64 Echo (ping) request  id=0x0002, seq=15/3840, ttl=255 (reply in 47)
192.168.75.2    44.10.2.3       ICMP    64 Echo (ping) reply    id=0x0002, seq=15/3840, ttl=60 (request in 46)
44.10.2.3       192.168.75.2    ICMP    64 Echo (ping) request  id=0x0002, seq=16/4096, ttl=255 (reply in 49)
192.168.75.2    44.10.2.3       ICMP    64 Echo (ping) reply    id=0x0002, seq=16/4096, ttl=60 (request in 48)
44.10.2.3       192.168.75.2    ICMP    64 Echo (ping) request  id=0x0002, seq=17/4352, ttl=255 (reply in 51)
192.168.75.2    44.10.2.3       ICMP    64 Echo (ping) reply    id=0x0002, seq=17/4352, ttl=60 (request in 50)
```

Figura 138. Prueba de conectividad simultánea en emulación Francia-Ucrania.



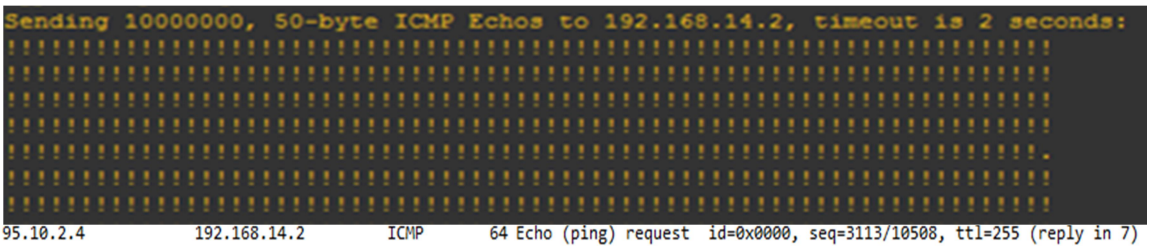
```
192.168.35.2    8.10.2.4        ICMP    64 Echo (ping) reply    id=0x0000, seq=1228/52228, ttl=56 (request in 874)
8.10.2.4        192.168.35.2    ICMP    64 Echo (ping) request  id=0x0000, seq=1229/52484, ttl=255 (reply in 877)
192.168.35.2    8.10.2.4        ICMP    64 Echo (ping) reply    id=0x0000, seq=1229/52484, ttl=56 (request in 876)
8.10.2.4        192.168.35.2    ICMP    64 Echo (ping) request  id=0x0000, seq=1230/52740, ttl=255 (reply in 879)
192.168.35.2    8.10.2.4        ICMP    64 Echo (ping) reply    id=0x0000, seq=1230/52740, ttl=56 (request in 878)
```

Figura 139. Prueba de conectividad simultánea en emulación Austria-Irlanda.




```
159.10.2.4      192.168.77.2    ICMP    64 Echo (ping) request  id=0x0000, seq=1237/54532, ttl=255 (reply in 4)
192.168.77.2    159.10.2.4      ICMP    64 Echo (ping) reply    id=0x0000, seq=1237/54532, ttl=60 (request in 3)
159.10.2.4      192.168.77.2    ICMP    64 Echo (ping) request  id=0x0000, seq=1238/54788, ttl=255 (reply in 6)
192.168.77.2    159.10.2.4      ICMP    64 Echo (ping) reply    id=0x0000, seq=1238/54788, ttl=60 (request in 5)
159.10.2.4      192.168.77.2    ICMP    64 Echo (ping) request  id=0x0000, seq=1239/55044, ttl=255 (reply in 8)
```

Figura 140. Prueba de conectividad simultánea en emulación Hungria-Letonia.



95.10.2.4	192.168.14.2	ICMP	64 Echo (ping) request	id=0x0000, seq=3113/10508, ttl=255 (reply in 7)
192.168.14.2	95.10.2.4	ICMP	64 Echo (ping) reply	id=0x0000, seq=3113/10508, ttl=61 (request in 6)
95.10.2.4	192.168.14.2	ICMP	64 Echo (ping) request	id=0x0000, seq=3114/10764, ttl=255 (reply in 9)
192.168.14.2	95.10.2.4	ICMP	64 Echo (ping) reply	id=0x0000, seq=3114/10764, ttl=61 (request in 8)
95.10.2.4	192.168.14.2	ICMP	64 Echo (ping) request	id=0x0000, seq=3115/11020, ttl=255 (reply in 11)

Figura 141. Prueba de conectividad simultánea en emulación Noruega-Estonia.

Como observamos en las figuras anteriores los nodos que quedaron deshabilitados en esta prueba fueron los de Alemania y Suiza, mientras que el nodo de Austria tuvo pérdidas de paquetes. El resto permanecieron con comunicación constante.

5.4. Gestión de GEANT en emulación

Dada la importancia de la red avanzada GEANT para las NREN europeas es indispensable mantener la disponibilidad y eficiencia en cada uno de sus PoP. Para lograrlo proponemos implementar una herramienta de gestión que nos permita monitorear de manera constante y segura en cada uno de los puntos de la red. Utilizaremos software basado en el protocolo SNMP para mantener comunicación constante entre la estación de gestión de red y los agentes configurados en cada uno de los PoP de GEANT. Una vez implementadas las configuraciones del protocolo SNMP en la parte de la metodología, comprobaremos las tres funciones básicas del protocolo SNMP; *Get, Next y Traps.*

5.4.1. Mensajes de Alerta en la gestión de GEANT (*Traps*)

Una vez agregado el agente configurado del router de UK a la consola de *SNMP Free Manager*, el protocolo SNMP mantendrá comunicación continua entre el agente y la estación de gestión por medio de mensajes de alerta llamados *Traps,* los cuales informan la actividad y acontecimientos relacionados con el equipo gestionado, como se muestra en la figura142.

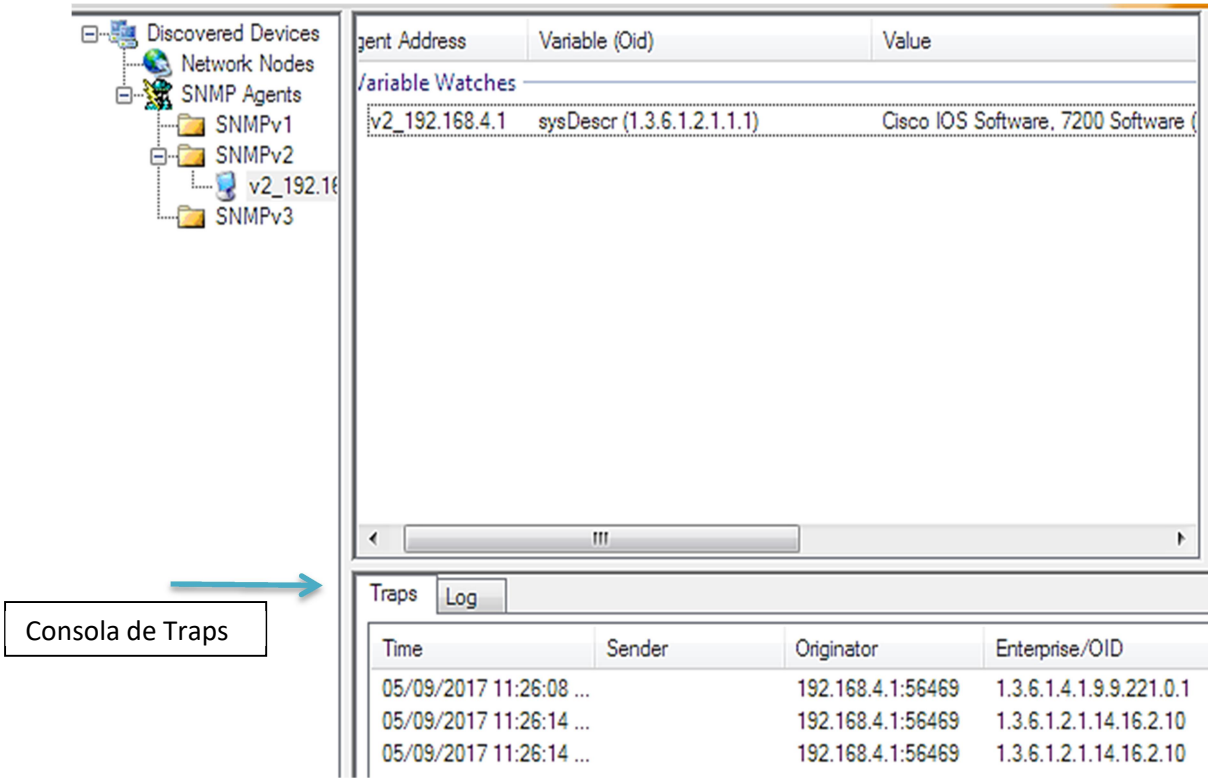


Figura 142. Consola de mensajes Traps.

5.4.2. Prueba de mensajes de tipo Trap en emulación de GEANT

Una vez configurados los agentes de Alemania e Israel y vinculados a *Power SNMP Free Manager*, deshabilitaremos alguno de los dos para comprobar que el protocolo por medio de un mensaje de tipo *Trap*, el cual enviara información del enlace, deshabilitado. Verificamos que la red 192.168.4.1 pertenece a la red de Alemania como se muestra en la figura 143.

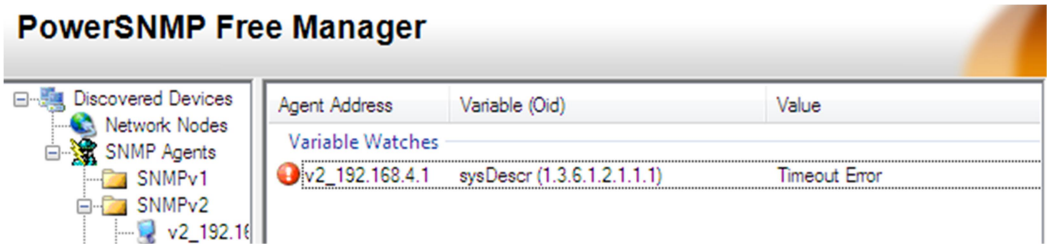


Figura 143. Notificación de SNMP del router de Alemania.

En el área de Traps de *Power SNMP Free Manager*, encontramos la OID que indica el la caída del router de Alemania como se muestra en la figura 144.

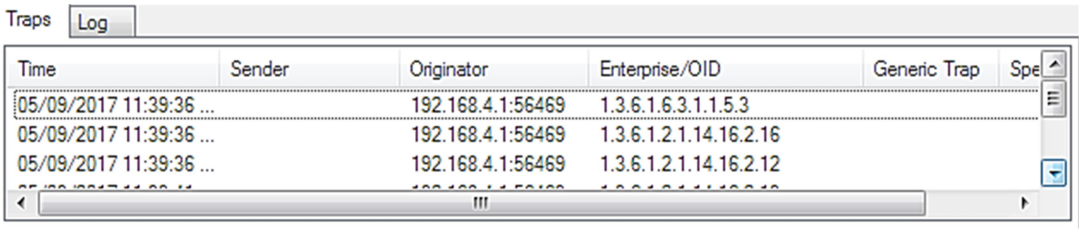


Figura 144. Notificación de Traps.

Verificando la OID con la herramienta de cisco *SNMP Object Navigator*, traducimos la OID de la trap marcada con fecha 04/02/2018 y hora 11:38:57, indica el enlace caído como se muestra en la figura 145.

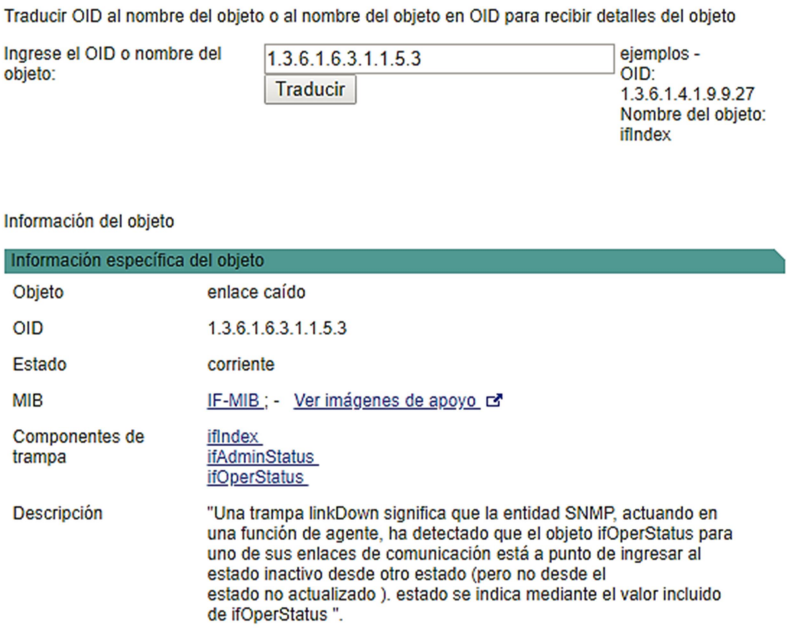


Figura 145. OID: 1.3.6.1.6.3.1.1.5.3. Estado Link Down

Después de detectar el router deshabilitado, el agente manda información por medio de otras traps para informar la actividad del protocolo OSPF ante la interfaz caída. Por medio de la OID: 1.3.6.1.2.1.14.16.2.2, identificado como objeto OSPF-Trap-MIB. Esta nos indica que hubo un cambio en un router vecino de la red, especificando que solo hay comunicación unidireccional, como se muestra en la figura 146.

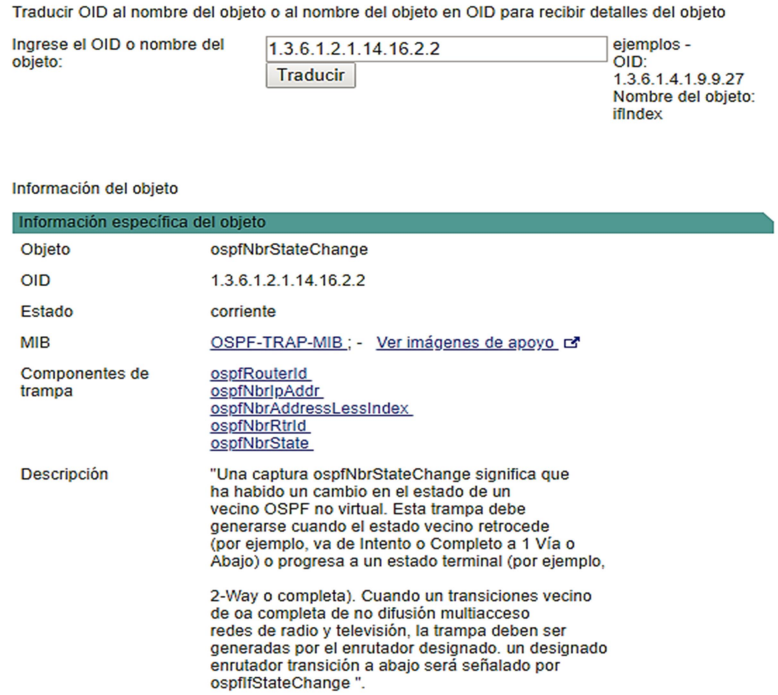


Figura 146. OID: Estado OspfNbr (Cambio de estado).

Siguiendo el orden de llegada de traps, detectamos la trap con OID: 1.3.6.1.2.1.14.16.2.16, que nos indica un cambio en una interfaz configurada con OSPF como se muestra en la figura 147.

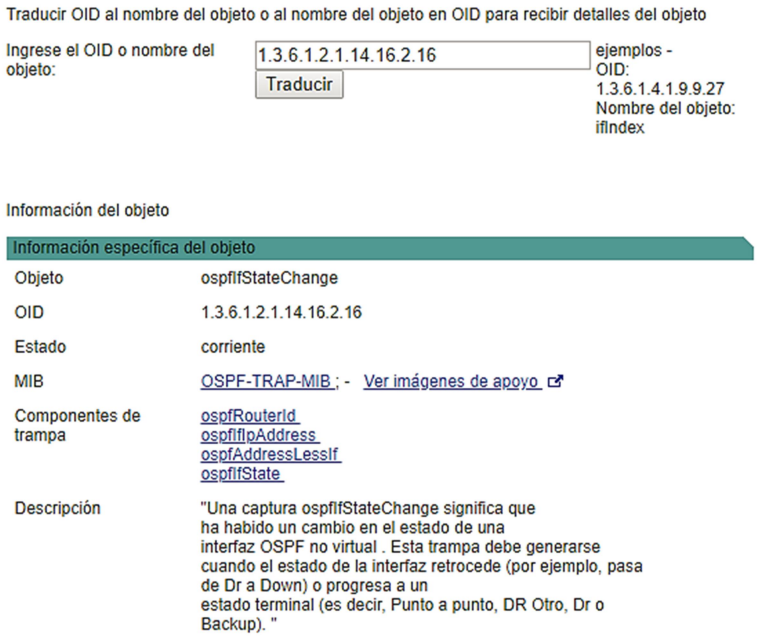


Figura 147. OID, *State Change* OSPF.

Por ultimo analizamos la trap con OID: 1.3.6.1.2.1.14.16.2.12 que nos informa de la generación de un mensaje de tipo LSA para informar a los demás routers del estado de la interfaz y actualizar sus tablas de enrutamiento OSPF, como lo vimos en la teoría y como lo muestra la figura 148.

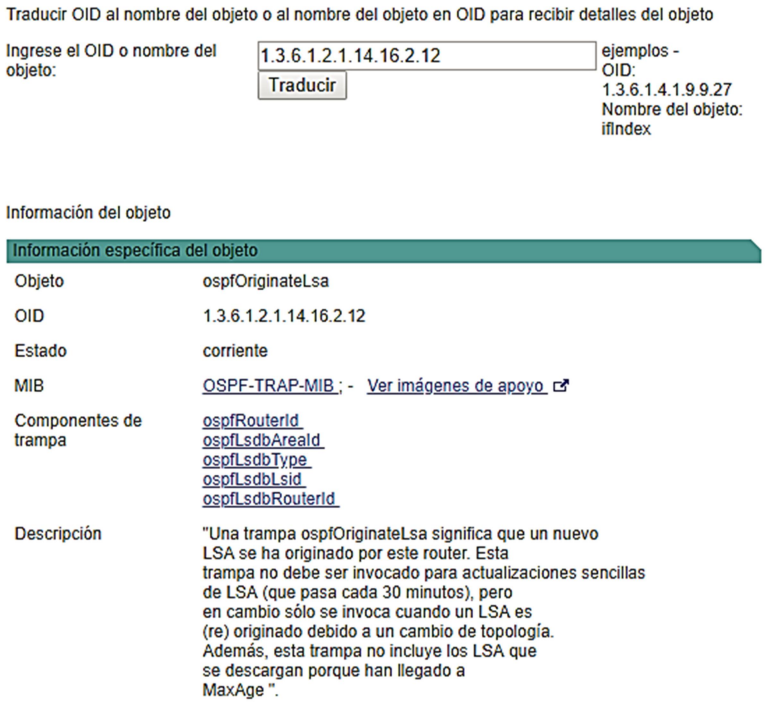


Figura 148. Estado Ospf se origina Una LSA.

5.4.3. Mensajes de tipo *Get* SNMP en la gestión de GEANT

En este apartado nos enfocaremos a la demostración del funcionamiento del protocolo SNMP por medio de los mensajes de tipo Get y Next para solicitar o modificar información de los equipos gestionados por medio de los agentes SNMP. Para este proceso utilizáremos el software *MIB Browser* como se muestra en la figura 149.

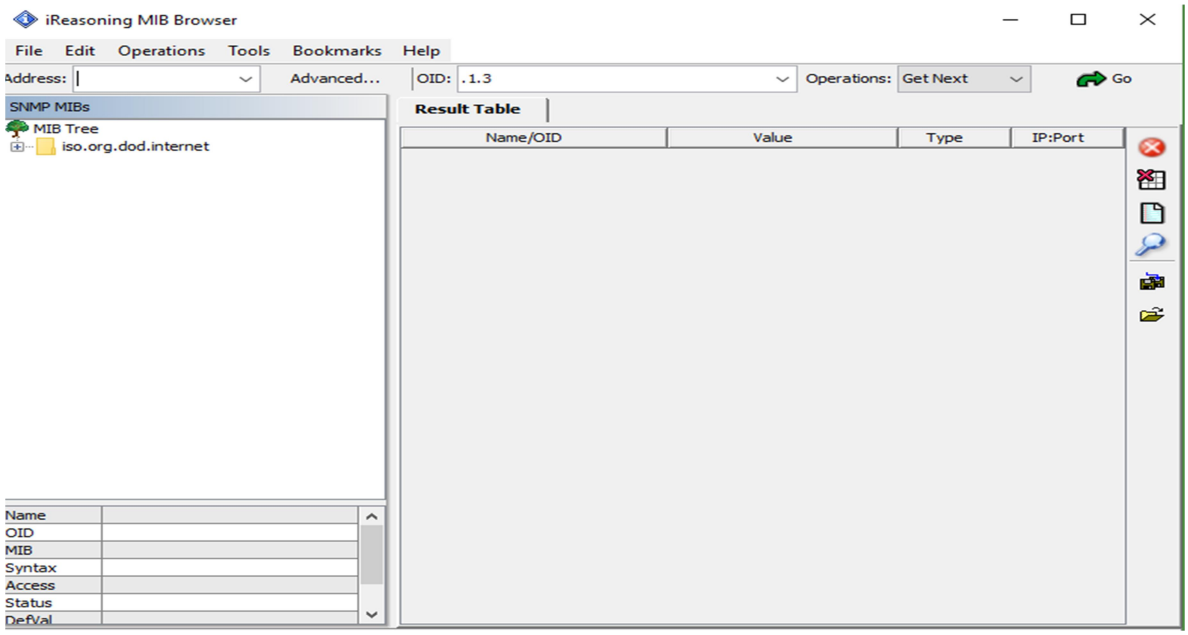


Figura 149. MIB Browser.

El siguiente paso es agregar los agentes configurados de GEANT en el MIB Browser. Agregamos la ruta de red donde se ubica el agente y damos click en *Advanced* para configurar la comunidad y la versión del protocolo como se muestra en la figura 150.

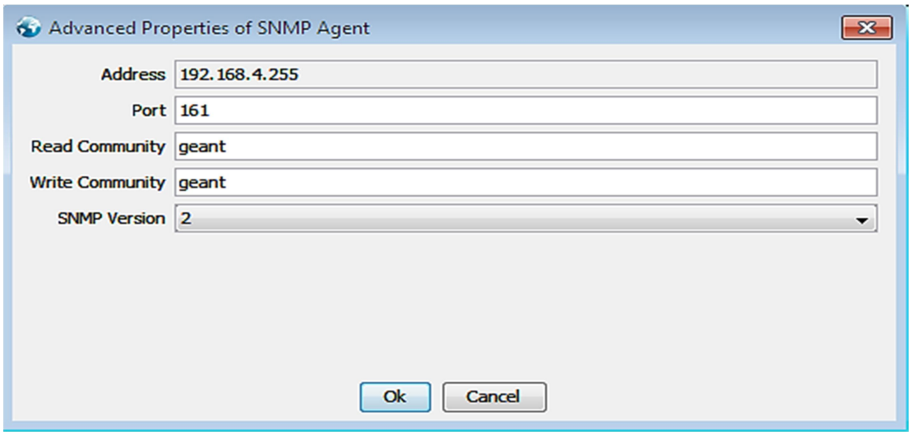


Figura 150. Configurando Agente a MIB browser.

El siguiente paso es llegar al objeto *Sysname* y solicitar la información de ese objeto, con la operación "*Get*", es decir, solicitar el nombre del router gestionado, como se muestra en la figura 151.

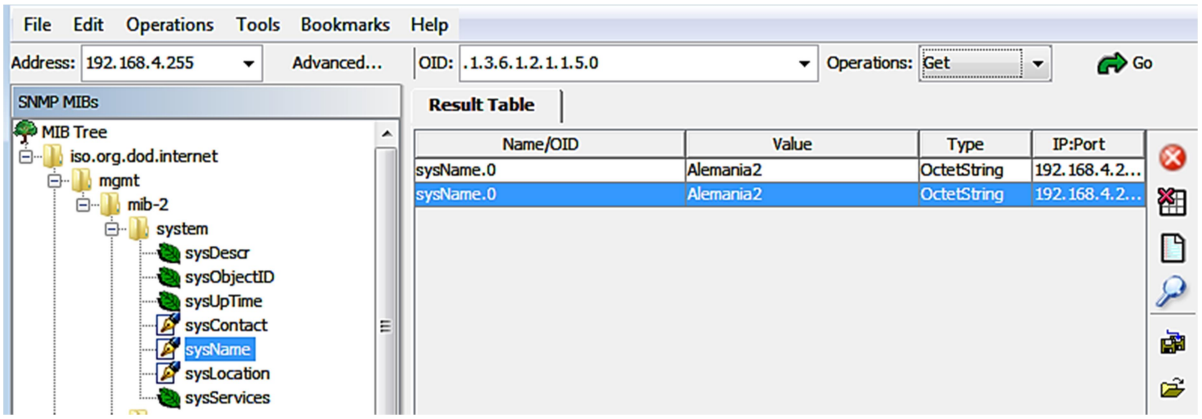


Figura 151. Mensaje Get SNMP para agente del router de Alemania.

5.4.4. Mensajes de tipo *Get* SNMP: Solicitud de tabla de enrutamiento

Después de haber solicitado el nombre del router de Alemania2, podemos buscar de manera específica los objetos del árbol MIB y modificar sus valores. En este caso solicitaremos información de la tabla de enrutamiento del router de Alemania por medio del mensaje *Get* como se muestra en la figura 152.

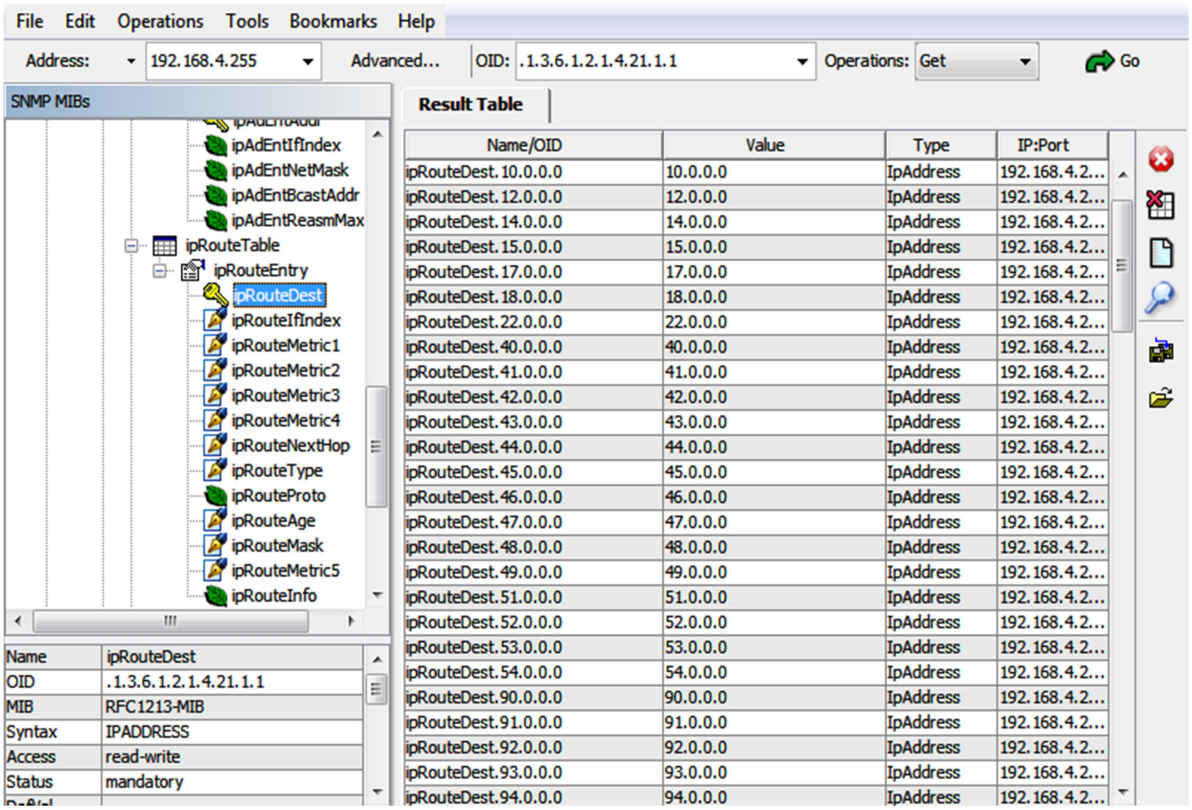


Figura 152. Tabla de enrutamiento del router de Alemania.

Para corroborar la tabla de enrutamiento nos basta con comparar la tabla de la figura 152 con la tabla de la figura 106.

5.4.5. Mensaje de tipo *Set* SNMP: Modificación de Valores

Por último comprobaremos el funcionamiento del mensaje *Set* en la emulación con el fin de modificar el valor del nombre del router de Alemania, para lo cual seleccionamos la OID del objeto que representa el valor del nombre del router y elegimos la operación *Set*. Realizado este paso seleccionamos en la ventana *SNMP Set,* el tipo de datos a cambiar y en el campo *Value* ingresamos el nuevo valor del nombre del router como se muestra en la figura 153.

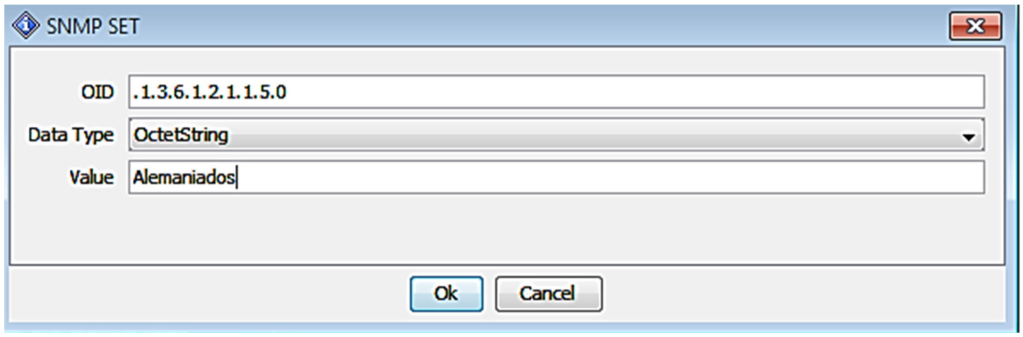


Figura 153. Mensaje Set SNMP.

Sin embargo, cuando enviamos el mensaje *Set*, se generó un mensaje de error indicando el acceso negado como se muestra en la figura 154.

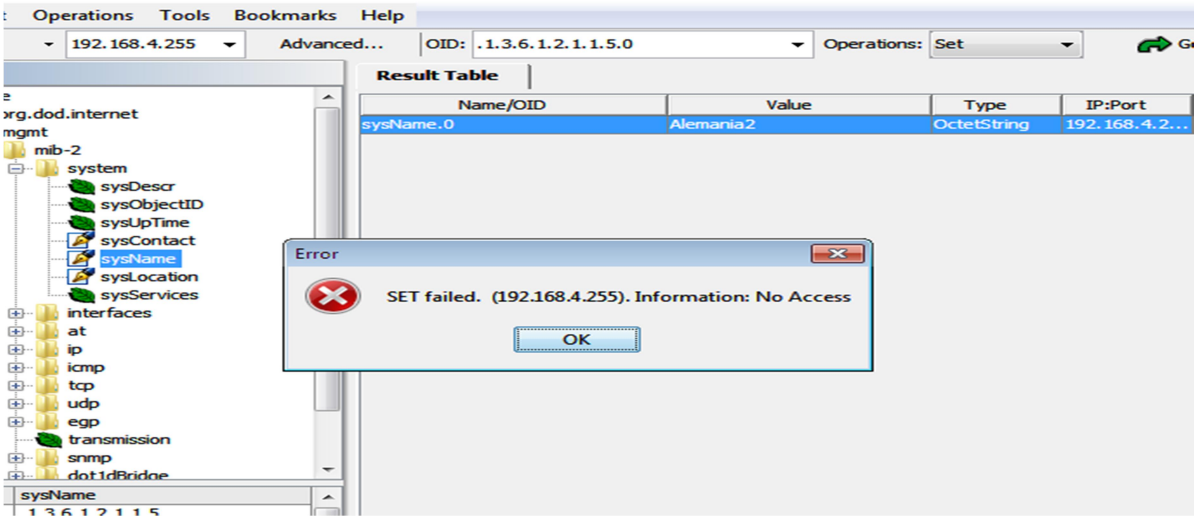


Figura 154. Fallo de mensaje *Set*.

Esto sucedió porque en la configuración de SNMP en GNS3 solo configuramos la opción *Get* como medida de seguridad para evitar una gestión no deseada. Por lo que necesitamos configurar la operación *Set* en nuestro agente en el router de Alemania como se muestra en la figura 155.

```
Alemania2 (config)#
Alemania2 (config)# snmp-server community geant rw SNMP_ACL
Alemania2 (config)# snmp-server host 192.168.4.2 version 2c geant
Alemania2 (config)# ip access-list standard SNMP_ACL
Alemania2 (config-std-nacl)# permit 192.168.4.2
Alemania2 (config-std-nacl)# exit
Alemania2 (config)#
```

Figura 155. Configuración del agente del router de Alemania2.

Después de haber configurado el agente para la operación *Set*, nuevamente realizamos la operación para cambiar el valor del nombre del router como se muestra en la figura 156-57.

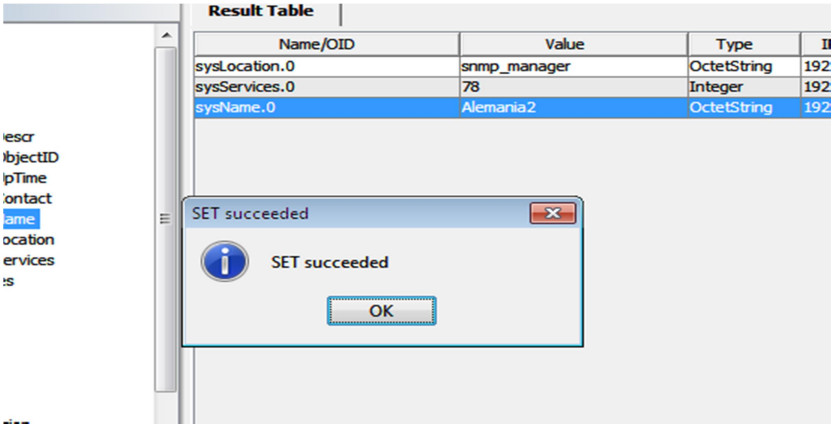


Figura 156. Mensaje Set con éxito.

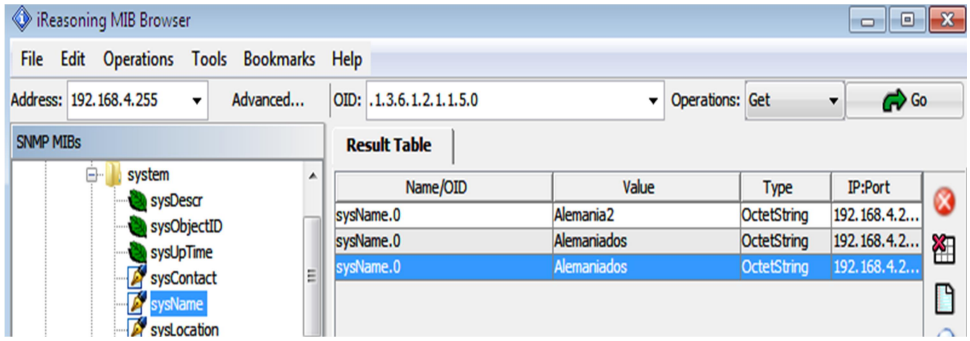


Figura 157. Mensaje *Set SNMP* con éxito

Finalizamos satisfactoriamente la prueba del mensaje *Set*, cambiando el valor del nombre del router de Alemania2 al valor “Alemaniados” como se muestra en la figura 158.

5.4.6. Monitoreo de mensajes SNMP con Wireshark

En esta sección expondremos el monitoreo que realizamos a la interface de red de Alemania. Wireshark detecta cada mensaje que pasa por la interfaz, observamos todos los mensajes que se generaron cuando deshabilitamos la interfaz de Alemania para comprobar los comportamientos de los protocolos OSPF y SNMP como se muestra en la figura 158.

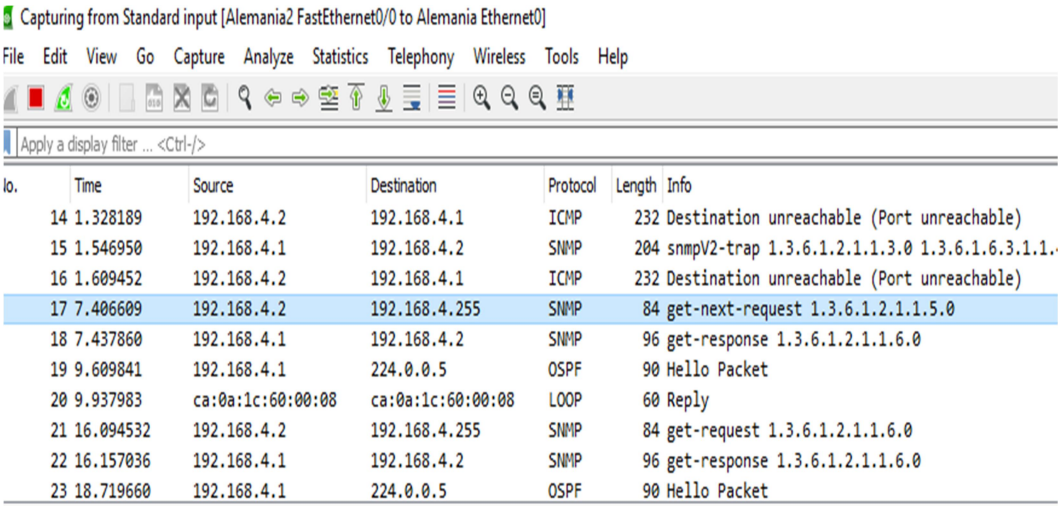


Figura 158. Monitoreo de Wireshark

El mensaje de tipo ICMP que se encarga de verificar errores en la conectividad de la red, en el monitoreo de Wireshark nos indica que la interfaz esta deshabilitada, el protocolo SNMP por medio de traps nos indica el evento sucedido y el protocolo OSPF con paquetes *Hello* trata de hacer adyacencia con la interfaz de red deshabilitada. Analizando el paquete de SNMP, este nos desglosa una especie de formato en donde nos indica el tipo de versión del protocolo, el tipo de mensaje, la comunidad que en este caso es "*geant*" y el campo para la detección de errores como se muestra en la figura 159.

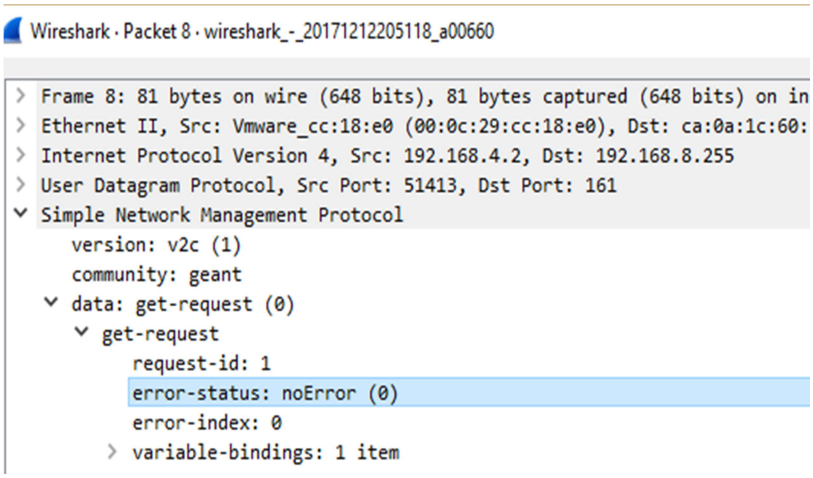


Figura 159. Formato de mensaje SNMP v2 en Wireshark.

5.4.7. Prueba de mensajes de tipo Trap en GEANT con SNMP v3

En esta sección se realizaron las mismas pruebas que se hicieron con SNMP v2 en la cual se deshabilitó el router de Alemania para que el software de gestión nos informara del suceso por medio de *Traps* como se muestra en la figura 160.

Figura 160. Router de Alemania deshabilitado.

El software envió la *Trap* correspondiente indicando que el enlace se deshabilito y mandó la OID "1.3.6.1.6.3.1.1.5.3" como se muestra en las figuras 161.

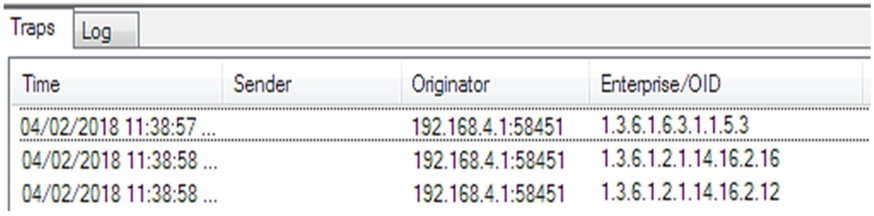


Figura 165. Traps con SNMP v3.

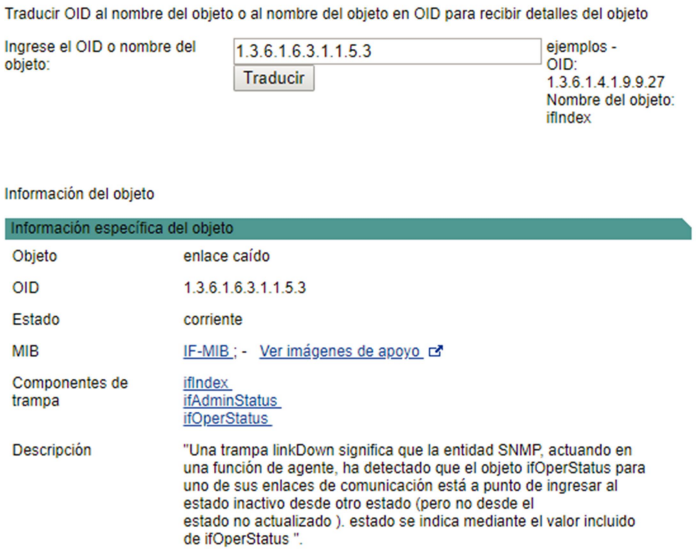


Figura 161. OID *Link Down* con *SNMP Object Navigator*

Posteriormente las siguientes OID de la figura 165, indican el cambio de estado de la interfaz de enlace y actualizaciones en las LSA de OSPF como se comprobó en la prueba con SNMP v2.

5.4.8. Mensajes de tipo *Get y Set* en GEANT con SNMP v3

Al realizar las pruebas de los mensajes de tipo *Get y Next* como lo hicimos anteriormente en SNMP v2 con el software MIB Browser nos fue imposible concluir la operación debido a que la versión del software nos limitó el uso de SNMP v3 por permisos de licencia como se muestra en la figura 162.

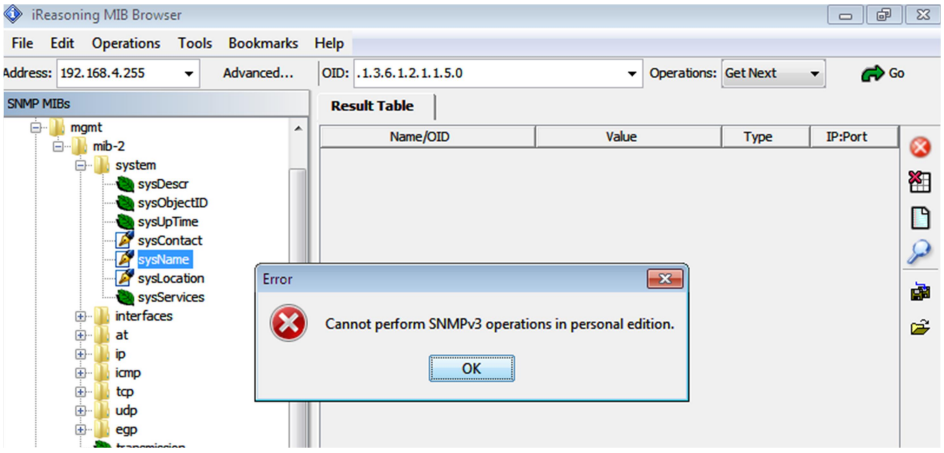


Figura 162. SNMP v3 en MIB Browser.

Capítulo 6

Conclusiones

Pruebas de conectividad con RIP v2 y OSPF v2 en Simulación

En el proceso de simulación, la configuración de los protocolos de enrutamiento se realizó de manera satisfactoria, usando el protocolo IPv4 e implementando de manera didáctica el uso de clases en las direcciones de red, sin embargo, el uso de un router de tipo Backbone en la simulación no fue posible, ya que Packet tracer solo ofrece routers de distribución y de tipo genérico, con interfaces Giga Ethernet, con un máximo de 10 interfaces. Por otro lado, el simulador solo nos permitió configurar como máximo 1 Gb de ancho de banda en cada interfaz, se comprobó conectividad vía CLI y GUI, comprobando los procesos con el comando *Ping*. También, se realizaron pruebas de redundancia y resilencia en la red aplicando RIP v2 y OSPF v2. Otra prueba que se realizó en simulación fue probar conectividad desde un mismo origen, destino, con RIP v2 y OSPF v2. En esta prueba observamos que el algoritmo de RIP v2 como el de OSPF v2, eligieron la misma ruta, sin alguna diferencia, considerando que esta ruta fue registrada con el número máximo de saltos por parte de RIP v2. No obstante realizamos pruebas de conectividad simultánea entre varios PoP´s de la red deshabilitando el router de Alemania, para probar redundancia en la red con RIP v2 y OSPF v2. Al comparar los datos obtenidos en ambas pruebas en OSPF v2, hubo un menor número de paquetes perdidos y respecto a la latencia el retardo de los paquetes fue menor en comparación con RIP v2 ya que, el tiempo de retado de paquetes en RIP v2 fue superior a los 2000 milisegundos mientras que en OSPF v2 se registraron retardos de paquetes de un máximo de 11 milisegundos.

Pruebas de Gestión con SNMP v2 en simulación

La configuración del protocolo de gestión SNMP en la simulación fue exitosa y con limitantes, respecto a los mensajes de tipo *Trap* y a la versión 3 del protocolo, ya que, no reconoció los comandos para configurar SNMP v3. No obstante, logramos implementar los mensajes de tipo *Get y Set*, comprobando las funciones de solitud y modificación de valores en los equipos gestionados y con la aplicación *SNMP Object Navigator* de cisco, comprobamos el significado de las OID del árbol MIB. La desventaja que tiene el simulador es que no permite configurar los mensajes de tipo *Trap* para SNMP v2 y el MIB Browser del Host configurado, no cuenta con una área de monitoreo en tiempo real de los equipos gestionados para recibir mensajes de alerta (*Trap*).

Pruebas de conectividad con OSPF v2 en Emulación

El proceso de emulación de la red GEANT se realizó de manera satisfactoria, configurando y probando conectividad de manera exitosa. En comparación con la simulación, se implementaron IOS reales del router C7200 de cisco, es decir, logramos utilizar routers de tipo Backbone, usando la virtualización de GNS3, con un máximo de 8 interfaces Giga Ethernet por router. Sin embargo, los routers que usamos en simulación nos ofrecieron más interfaces, razón que nos obligó a aumentar el número de routers en la emulación, para cubrir los enlaces de los PoP´s de Austria y Hungria. El emulador solo nos ofreció interfaces CLI para la configuración y comprobación de conectividad, mediante los routers, las VPCS y las MV. No obstante, para probar conectividad y registrar el proceso de cada paquete, se utilizó *Wireshark*, este nos permitió capturar los paquetes que viajan en toda la red como los de tipo ARP, ICMP, OSPF y SNMP. Para las pruebas de conectividad desde una máquina virtual, se tuvo que configurar el emulador GNS3 con el software de virtualización VM Ware, configurar la máquina virtual con el sistema operativo Windows 7, configurar su red con los datos del host de Alemania y desactivar el firewall de manera general. Dada la virtualización de todos los routers de la red y el uso de máquinas virtuales los procesos en la red fueron más lentos, sin embargo no hubo problemas en su funcionamiento.

Pruebas de Gestión con SNMP v2 y SNMP v3 en Emulación

Por otro lado la configuración del protocolo SNMP v2 y SNMP v3 se realizó de manera satisfactoria, en comparación con la simulación logramos configurar, traps y SNMP v3 con autentificación y cifrado. Para el proceso de gestión en la emulación fue necesario el uso de máquinas virtuales que nos brindaron una plataforma de gestión, con software basado en SNMP. El software *Power SNMP Free Manager y MIB Browser,* nos permitieron realizar pruebas de monitoreo y gestión en la emulación de GEANT, utilizando mensajes de tipo *Get, Next y Traps.* Las pruebas de monitoreo y gestión usando SNMP v2 se realizaron de manera satisfactoria logrando demostrar cada una de las funciones de sus tres tipos de mensajes. Por otro lado con SNMP v3 se logró implementar autentificación y cifrado entre la estación de gestión y sus entidades, comprobando las funciones de los mensajes de tipo *Get* y *Trap* usando *Power SNMP Free Manager,* pero, no se pudo comprobar el funcionamiento de los mensajes de tipo *Set* por falta de permisos para SNMP v3 en ambos programas. Concluimos con base en la implementación de SNMP en sus versiones 2 y 3, que es un protocolo de gestión bastante funcional y fácil de implementar ya que su configuraciones en los agentes, entidades y estaciones de gestión, no son complejas, a cambio logramos tener una herramienta que monitorea todo el tiempo el estado de los equipos de la red de forma física y lógica. No obstante es necesario realizar más pruebas en redes con

mayor tráfico para ver su rendimiento y funcionamiento máximo del protocolo y determinar las mejores estrategias para su uso.

Para los procesos de simulación y emulación de GEANT, se utilizó un equipo portátil de la marca *Sony*, con procesador Intel Core(TM) i5-2410M, CPU 2.30 GHz- 2.30 GHz., con memora RAM de 12.0 GB y disco de estado sólido de 128 GB. Como comparativa entre la simulación y emulación se registró el rendimiento del equipo físico en ambos procesos y los resultados fueron los siguientes:

Simulación:

CPU: 10.4%

Memoria: 240.3 MB de 12 Gb= 2.025%

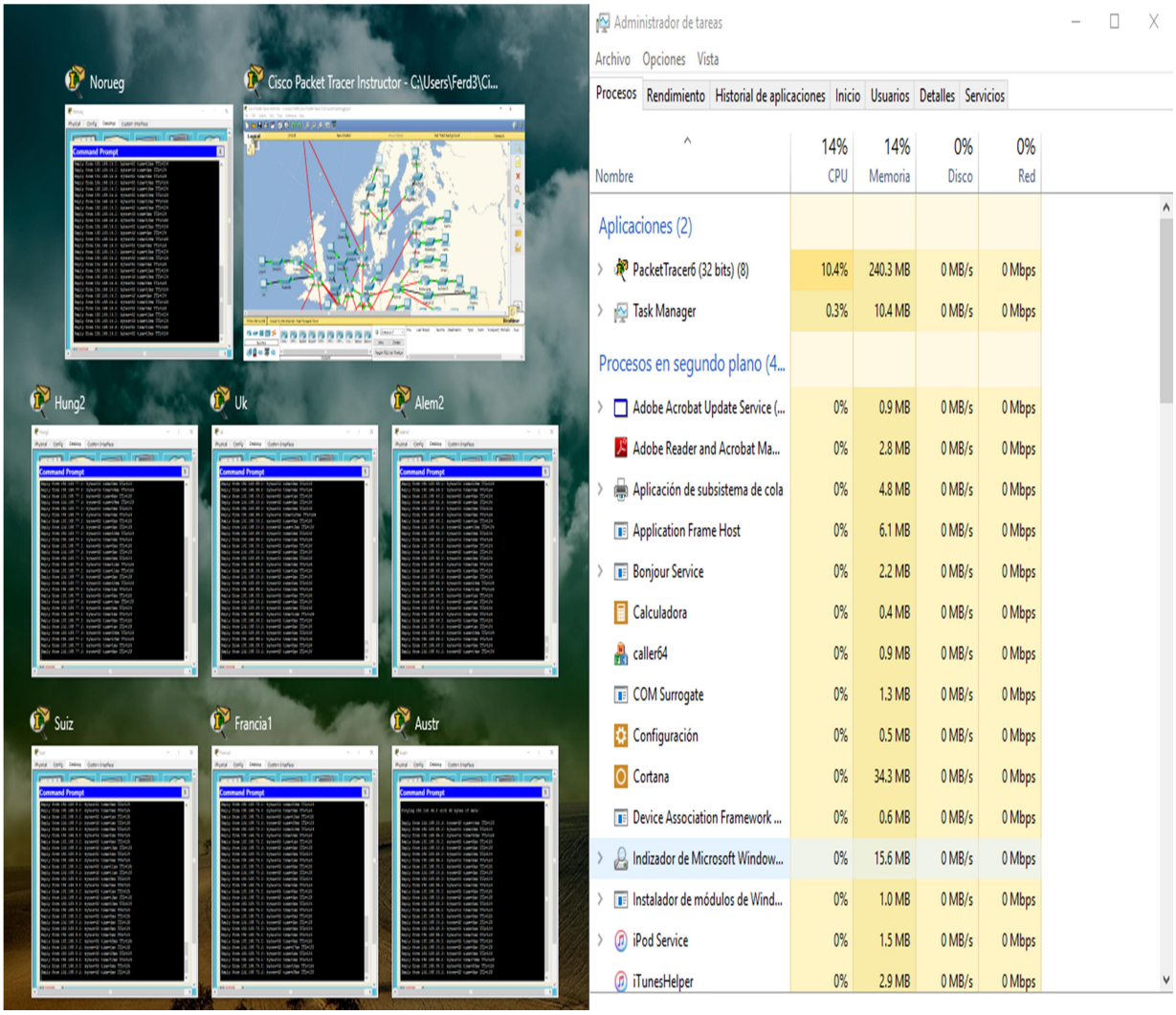


Figura 163. Rendimiento de CPU y Memoria del equipo físico en simulación.

Emulación:

CPU: 100%

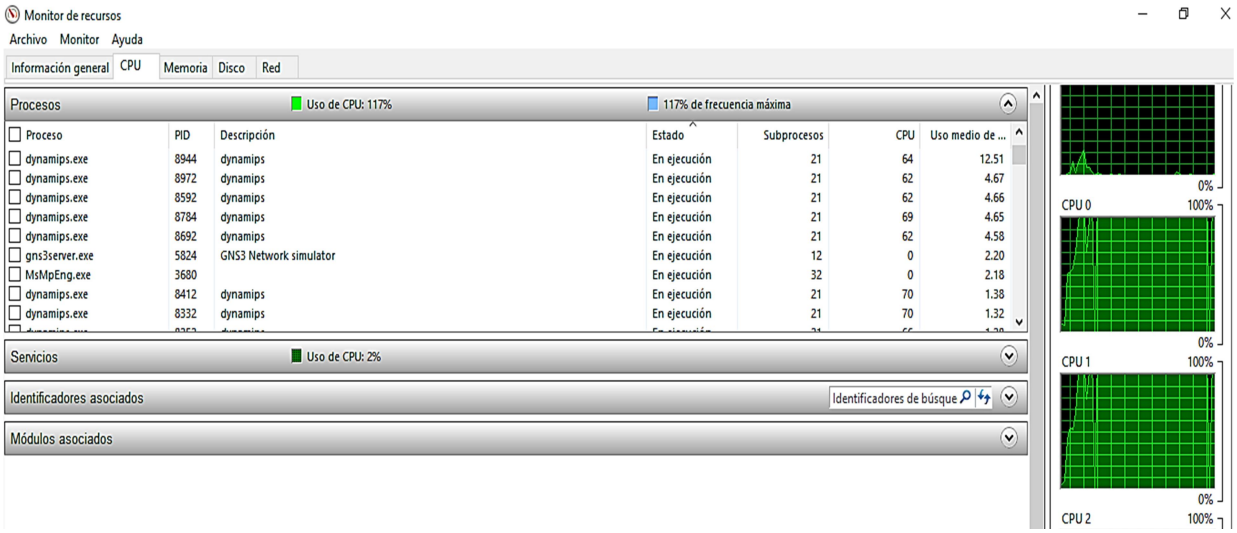


Figura 164. Rendimiento de CPU del equipo físico en emulación.

RAM: 55%

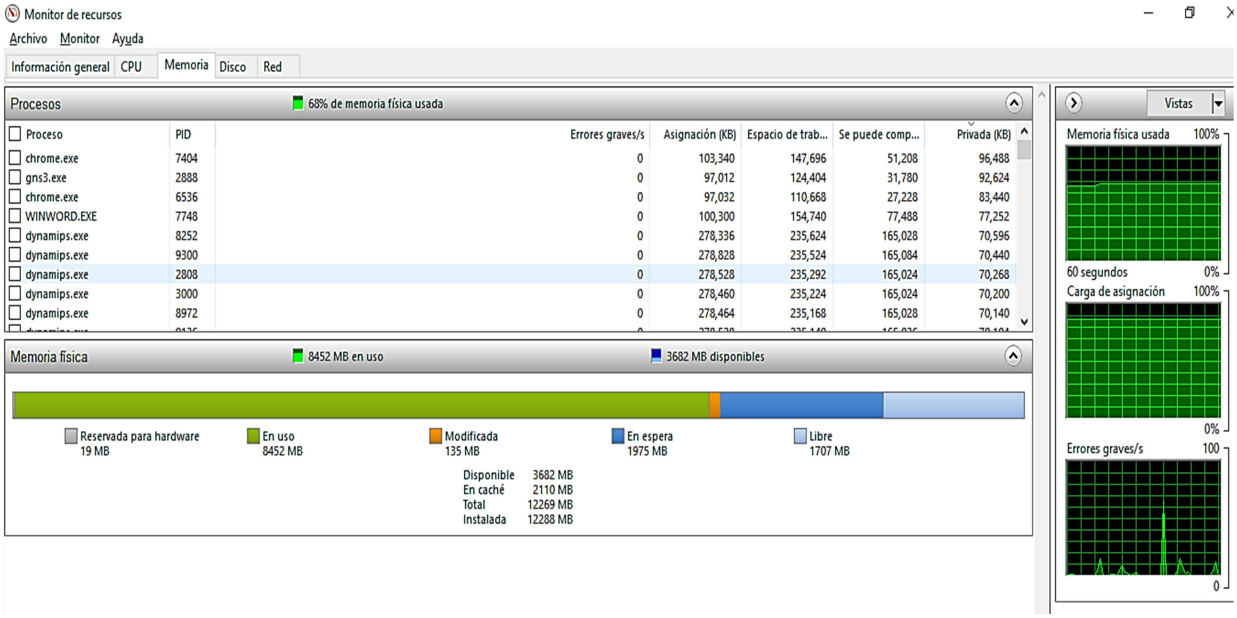


Figura 165. Registro de uso de RAM en equipo físico en emulación.

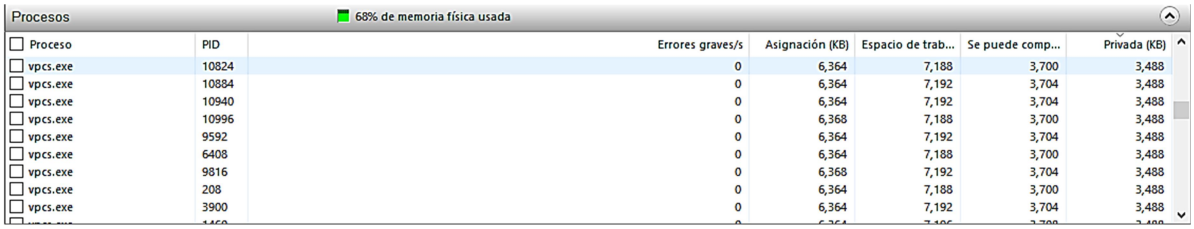


Figura 166. Registro de uso de RAM de las VPCS en emulación.

En la emulación se usaron un total de 83 elementos virtuales, de los cuales, 44 fueron routers, 38 VPCS y 1 Máquina Virtual. El uso promedio de RAM, en cada router fue de 270MB, pero estos se configuraron para que solo ocuparan la mitad de la memoria asignada. Para las VPCS se registró un uso de RAM de 6 MB y para la máquina virtual un uso de 512 MB, dando un total de uso en la emulación de 6545 MB, es decir, 55 % de uso de RAM.

Después de haber analizado el funcionamiento del Backbone de GEANT, en simulación y emulación por medio de software, deducimos que es posible simular y emular su capa de transmisión, incluso, probar conectividad y gestión. No obstante, existieron limitantes, que solo nos permitieron crear un ambiente didáctico de redes para probar el funcionamiento de los protocolos de enrutamiento y gestión. Sin embargo, es importante decir que a pesar de ello se pudo crear y probar con éxito una red de 42 routers con interfaces Giga Ethernet, realizando pruebas de conectividad simultánea, además de realizar pruebas de gestión basadas en SNMP. Por otro lado fue aceptable el uso de recursos del simulador ya que funcionó perfectamente permitiendo realizar todas las pruebas necesarias para este trabajo ocupando el 10% de CPU y aproximadamente el 2% de RAM.

Por otro lado en la emulación el gasto de recursos en el CPU del equipo físico, fue del 100 %, mismo que, repercutió en los procesos, ya que estos fueron mucho más lentos, sin embargo no existieron problemas en las pruebas de red. Se pudo comprobar el funcionamiento de OSPF v2, SNMP v2 y v3, habilitar traps, cifrado y autentificación SNMP. La emulación se acercó más a un modelo real ya que cada router empleado, usó los recursos reales del equipo físico y su funcionamiento fue idéntico a un router real. Así mismo, el uso de una máquina virtual, configurada con software autentico, permitió una gestión en tiempo real de una red con el funcionamiento de routers de tipo Backbone.

De esta manera concluimos que el simulador Packet Tracer y el emulador GNS3, son herramientas para el aprendizaje de redes, que nos ayudan a vincular los conceptos teóricos de redes, con modelos prácticos aproximados al funcionamiento de una red. Sin embargo, es necesario buscar nuevas herramientas o desarrollar emuladores más potentes e implementarlos en computadoras con mayores recursos, preparadas para soportar virtualización de varios equipos y lograr igualar el funcionamiento real de una red de Backbone como GEANT, para proponer nuevas estrategias de red que nos permitan ofrecer redes de alto desempeño y alta disponibilidad.

Apéndice A: Prueba de conectividad usando el protocolo RIP v2

```
PC>ping 192.168.2.2

Pinging 192.168.2.2 with 32 bytes of data:

Reply from 192.168.2.2: bytes=32 time=7ms TTL=128
Reply from 192.168.2.2: bytes=32 time=5ms TTL=128
Reply from 192.168.2.2: bytes=32 time=6ms TTL=128
Reply from 192.168.2.2: bytes=32 time=5ms TTL=128

Ping statistics for 192.168.2.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 5ms, Maximum = 7ms, Average = 5ms

PC>ping 192.168.4.2

Pinging 192.168.4.2 with 32 bytes of data:

Request timed out.
Reply from 192.168.4.2: bytes=32 time=0ms TTL=126
Reply from 192.168.4.2: bytes=32 time=0ms TTL=126
Reply from 192.168.4.2: bytes=32 time=0ms TTL=126

Ping statistics for 192.168.4.2:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

Figura 1. Prueba de conectividad de Alemania 2 a Alemania

```
PC>ping 192.168.15.2

Pinging 192.168.15.2 with 32 bytes of data:

Request timed out.
Reply from 192.168.15.2: bytes=32 time=0ms TTL=125
Reply from 192.168.15.2: bytes=32 time=10ms TTL=125
Reply from 192.168.15.2: bytes=32 time=0ms TTL=125

Ping statistics for 192.168.15.2:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 10ms, Average = 3ms

PC>ping 192.168.14.2

Pinging 192.168.14.2 with 32 bytes of data:

Request timed out.
Reply from 192.168.14.2: bytes=32 time=0ms TTL=125
Reply from 192.168.14.2: bytes=32 time=0ms TTL=125
Reply from 192.168.14.2: bytes=32 time=0ms TTL=125

Ping statistics for 192.168.14.2:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

Figura 2. Prueba de conectividad Ping a Dinamarca y Estonia.

```
PC>ping 192.168.43.2

Pinging 192.168.43.2 with 32 bytes of data:

Request timed out.
Reply from 192.168.43.2: bytes=32 time=0ms TTL=124
Reply from 192.168.43.2: bytes=32 time=0ms TTL=124
Reply from 192.168.43.2: bytes=32 time=0ms TTL=124

Ping statistics for 192.168.43.2:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms

PC>ping 192.168.80.2

Pinging 192.168.80.2 with 32 bytes of data:

Request timed out.
Reply from 192.168.80.2: bytes=32 time=0ms TTL=122
Reply from 192.168.80.2: bytes=32 time=11ms TTL=122
Reply from 192.168.80.2: bytes=32 time=9ms TTL=122

Ping statistics for 192.168.80.2:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 11ms, Average = 6ms
```

Figura 3. Prueba de conectividad Ping a España y Finlandia.

```
PC>ping 192.168.37.2

Pinging 192.168.37.2 with 32 bytes of data:

Request timed out.
Reply from 192.168.37.2: bytes=32 time=0ms TTL=125
Reply from 192.168.37.2: bytes=32 time=0ms TTL=125
Reply from 192.168.37.2: bytes=32 time=0ms TTL=125

Ping statistics for 192.168.37.2:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms

PC>ping 192.168.44.2

Pinging 192.168.44.2 with 32 bytes of data:

Request timed out.
Reply from 192.168.44.2: bytes=32 time=0ms TTL=125
Reply from 192.168.44.2: bytes=32 time=0ms TTL=125
Reply from 192.168.44.2: bytes=32 time=0ms TTL=125

Ping statistics for 192.168.44.2:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

Figura 4. Prueba de conectividad Ping a Francia y Francia 2.

```
PC>ping 192.168.32.2

Pinging 192.168.32.2 with 32 bytes of data:

Request timed out.
Reply from 192.168.32.2: bytes=32 time=1ms TTL=124
Reply from 192.168.32.2: bytes=32 time=0ms TTL=124
Reply from 192.168.32.2: bytes=32 time=0ms TTL=124

Ping statistics for 192.168.32.2:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 1ms, Average = 0ms

PC>ping 192.168.19.2

Pinging 192.168.19.2 with 32 bytes of data:

Request timed out.
Reply from 192.168.19.2: bytes=32 time=1ms TTL=124
Reply from 192.168.19.2: bytes=32 time=37ms TTL=124
Reply from 192.168.19.2: bytes=32 time=0ms TTL=124

Ping statistics for 192.168.19.2:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 37ms, Average = 12ms
```

Figura 5. Prueba de Conectividad Ping a Grecia y Croacia.

```
PC>ping 192.168.17.2

Pinging 192.168.17.2 with 32 bytes of data:

Request timed out.
Reply from 192.168.17.2: bytes=32 time=18ms TTL=124
Reply from 192.168.17.2: bytes=32 time=0ms TTL=124
Reply from 192.168.17.2: bytes=32 time=0ms TTL=124

Ping statistics for 192.168.17.2:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 18ms, Average = 6ms

PC>ping 192.168.35.2

Pinging 192.168.35.2 with 32 bytes of data:

Request timed out.
Reply from 192.168.35.2: bytes=32 time=0ms TTL=124
Reply from 192.168.35.2: bytes=32 time=0ms TTL=124
Reply from 192.168.35.2: bytes=32 time=10ms TTL=124

Ping statistics for 192.168.35.2:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 10ms, Average = 3ms
```

Figura 6. Prueba de conectividad Ping a Hungría 2 e Irlanda

```
PC>ping 192.168.8.2

Pinging 192.168.8.2 with 32 bytes of data:

Request timed out.
Reply from 192.168.8.2: bytes=32 time=0ms TTL=126
Reply from 192.168.8.2: bytes=32 time=1ms TTL=126
Reply from 192.168.8.2: bytes=32 time=1ms TTL=126

Ping statistics for 192.168.8.2:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 1ms, Average = 0ms

PC>ping 192.168.40.2

Pinging 192.168.40.2 with 32 bytes of data:

Request timed out.
Reply from 192.168.40.2: bytes=32 time=0ms TTL=124
Reply from 192.168.40.2: bytes=32 time=0ms TTL=124
Reply from 192.168.40.2: bytes=32 time=0ms TTL=124

Ping statistics for 192.168.40.2:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

Figura 7. Prueba de conectividad Ping a Israel e Islandia.

```
PC>ping 192.168.31.2

Pinging 192.168.31.2 with 32 bytes of data:

Request timed out.
Reply from 192.168.31.2: bytes=32 time=13ms TTL=125
Reply from 192.168.31.2: bytes=32 time=0ms TTL=125
Reply from 192.168.31.2: bytes=32 time=0ms TTL=125

Ping statistics for 192.168.31.2:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 13ms, Average = 4ms

PC>ping 192.168.73.2

Pinging 192.168.73.2 with 32 bytes of data:

Request timed out.
Reply from 192.168.73.2: bytes=32 time=0ms TTL=125
Reply from 192.168.73.2: bytes=32 time=0ms TTL=125
Reply from 192.168.73.2: bytes=32 time=11ms TTL=125

Ping statistics for 192.168.73.2:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 11ms, Average = 3ms
```

Figura 8. Prueba de Conectividad Ping a Italia y Lituania.

```
PC>ping 192.168.6.2

Pinging 192.168.6.2 with 32 bytes of data:

Request timed out.
Reply from 192.168.6.2: bytes=32 time=0ms TTL=126
Reply from 192.168.6.2: bytes=32 time=0ms TTL=126
Reply from 192.168.6.2: bytes=32 time=0ms TTL=126

Ping statistics for 192.168.6.2:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms

PC>ping 192.168.77.2

Pinging 192.168.77.2 with 32 bytes of data:

Request timed out.
Reply from 192.168.77.2: bytes=32 time=11ms TTL=124
Reply from 192.168.77.2: bytes=32 time=0ms TTL=124
Reply from 192.168.77.2: bytes=32 time=23ms TTL=124

Ping statistics for 192.168.77.2:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 23ms, Average = 11ms
```

Figura 9. Prueba de Conectividad Ping de a Luxemburgo y Letonia...

```
PC>ping 192.168.18.2

Pinging 192.168.18.2 with 32 bytes of data:

Request timed out.
Reply from 192.168.18.2: bytes=32 time=0ms TTL=123
Reply from 192.168.18.2: bytes=32 time=0ms TTL=123
Reply from 192.168.18.2: bytes=32 time=0ms TTL=123

Ping statistics for 192.168.18.2:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms

PC>ping 192.168.53.2

Pinging 192.168.53.2 with 32 bytes of data:

Request timed out.
Reply from 192.168.53.2: bytes=32 time=0ms TTL=123
Reply from 192.168.53.2: bytes=32 time=0ms TTL=123
Reply from 192.168.53.2: bytes=32 time=11ms TTL=123

Ping statistics for 192.168.53.2:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 11ms, Average = 3ms
```

Figura 10. Prueba de conectividad Ping a Montenegro y Macedonia

.

```
PC>ping 192.168.46.2

Pinging 192.168.46.2 with 32 bytes of data:

Request timed out.
Reply from 192.168.46.2: bytes=32 time=0ms TTL=124
Reply from 192.168.46.2: bytes=32 time=0ms TTL=124
Reply from 192.168.46.2: bytes=32 time=1ms TTL=124

Ping statistics for 192.168.46.2:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 1ms, Average = 0ms

PC>ping 192.168.3.2

Pinging 192.168.3.2 with 32 bytes of data:

Request timed out.
Reply from 192.168.3.2: bytes=32 time=11ms TTL=126
Reply from 192.168.3.2: bytes=32 time=12ms TTL=126
Reply from 192.168.3.2: bytes=32 time=0ms TTL=126

Ping statistics for 192.168.3.2:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 12ms, Average = 7ms
```

Figura 11. Prueba de Conectividad Ping a Malta y Holanda.

```
PC>ping 192.168.78.2

Pinging 192.168.78.2 with 32 bytes of data:

Request timed out.
Reply from 192.168.78.2: bytes=32 time=1ms TTL=124
Reply from 192.168.78.2: bytes=32 time=0ms TTL=124
Reply from 192.168.78.2: bytes=32 time=0ms TTL=124

Ping statistics for 192.168.78.2:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 1ms, Average = 0ms

PC>ping 192.168.5.2

Pinging 192.168.5.2 with 32 bytes of data:

Request timed out.
Reply from 192.168.5.2: bytes=32 time=2ms TTL=126
Reply from 192.168.5.2: bytes=32 time=1ms TTL=126
Reply from 192.168.5.2: bytes=32 time=0ms TTL=126

Ping statistics for 192.168.5.2:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 2ms, Average = 1ms
```

Figura 12. Prueba de conectividad Ping a Noruega y Polonia.

```
PC>ping 192.168.38.2

Pinging 192.168.38.2 with 32 bytes of data:

Request timed out.
Reply from 192.168.38.2: bytes=32 time=0ms TTL=124
Reply from 192.168.38.2: bytes=32 time=0ms TTL=124
Reply from 192.168.38.2: bytes=32 time=0ms TTL=124

Ping statistics for 192.168.38.2:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms

PC>ping 192.168.22.2

Pinging 192.168.22.2 with 32 bytes of data:

Request timed out.
Reply from 192.168.22.2: bytes=32 time=1ms TTL=124
Reply from 192.168.22.2: bytes=32 time=2ms TTL=124
Reply from 192.168.22.2: bytes=32 time=0ms TTL=124

Ping statistics for 192.168.22.2:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 2ms, Average = 1ms
```

Figura 13. Prueba de conectividad Ping a Portugal y Rumania.

```
PC>ping 192.168.21.2

Pinging 192.168.21.2 with 32 bytes of data:

Request timed out.
Reply from 192.168.21.2: bytes=32 time=0ms TTL=123
Reply from 192.168.21.2: bytes=32 time=0ms TTL=123
Reply from 192.168.21.2: bytes=32 time=0ms TTL=123

Ping statistics for 192.168.21.2:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms

PC>ping 192.168.74.2

Pinging 192.168.74.2 with 32 bytes of data:

Request timed out.
Reply from 192.168.74.2: bytes=32 time=31ms TTL=125
Reply from 192.168.74.2: bytes=32 time=0ms TTL=125
Reply from 192.168.74.2: bytes=32 time=0ms TTL=125

Ping statistics for 192.168.74.2:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 31ms, Average = 10ms
```

Figura 14. Prueba de conectividad Ping a serbia y Bielorrusia.

```
PC>ping 192.168.79.2

Pinging 192.168.79.2 with 32 bytes of data:

Request timed out.
Reply from 192.168.79.2: bytes=32 time=1ms TTL=123
Reply from 192.168.79.2: bytes=32 time=0ms TTL=123
Reply from 192.168.79.2: bytes=32 time=0ms TTL=123

Ping statistics for 192.168.79.2:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 1ms, Average = 0ms

PC>ping 192.168.33.2

Pinging 192.168.33.2 with 32 bytes of data:

Request timed out.
Reply from 192.168.33.2: bytes=32 time=0ms TTL=124
Reply from 192.168.33.2: bytes=32 time=0ms TTL=124
Reply from 192.168.33.2: bytes=32 time=0ms TTL=124

Ping statistics for 192.168.33.2:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

Figura 15. Prueba de conectividad Ping a Suecia y Eslovenia.

```
PC>ping 192.168.67.2

Pinging 192.168.67.2 with 32 bytes of data:

Request timed out.
Reply from 192.168.67.2: bytes=32 time=6ms TTL=123
Reply from 192.168.67.2: bytes=32 time=6ms TTL=123
Reply from 192.168.67.2: bytes=32 time=4ms TTL=123

Ping statistics for 192.168.67.2:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 4ms, Maximum = 6ms, Average = 5ms

PC>ping 192.168.20.2

Pinging 192.168.20.2 with 32 bytes of data:

Reply from 192.168.20.2: bytes=32 time=40ms TTL=123
Reply from 192.168.20.2: bytes=32 time=7ms TTL=123
Reply from 192.168.20.2: bytes=32 time=5ms TTL=123
Reply from 192.168.20.2: bytes=32 time=9ms TTL=123

Ping statistics for 192.168.20.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 5ms, Maximum = 40ms, Average = 15ms
```

Figura 16. Prueba de conectividad Ping a Moldavia y Eslovaquia.

```
PC>ping 192.168.9.2

Pinging 192.168.9.2 with 32 bytes of data:

Request timed out.
Reply from 192.168.9.2: bytes=32 time=0ms TTL=126
Reply from 192.168.9.2: bytes=32 time=1ms TTL=126
Reply from 192.168.9.2: bytes=32 time=0ms TTL=126

Ping statistics for 192.168.9.2:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 1ms, Average = 0ms

PC>ping 192.168.36.2

Pinging 192.168.36.2 with 32 bytes of data:

Request timed out.
Reply from 192.168.36.2: bytes=32 time=21ms TTL=125
Reply from 192.168.36.2: bytes=32 time=0ms TTL=125
Reply from 192.168.36.2: bytes=32 time=1ms TTL=125

Ping statistics for 192.168.36.2:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 21ms, Average = 7ms
```

Figura 17. Prueba de conectividad Ping a Turquía y UK.

```
PC>ping 192.168.75.1

Pinging 192.168.75.1 with 32 bytes of data:

Reply from 192.168.75.1: bytes=32 time=0ms TTL=253
Reply from 192.168.75.1: bytes=32 time=0ms TTL=253
Reply from 192.168.75.1: bytes=32 time=0ms TTL=253
Reply from 192.168.75.1: bytes=32 time=0ms TTL=253

Ping statistics for 192.168.75.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

Figura 18. Prueba de conectividad Ping a Ucrania.

Apéndice B: Prueba de conectividad usando OSPF en simulación.

```
PC>ping 192.168.4.2

Pinging 192.168.4.2 with 32 bytes of data:

Request timed out.
Reply from 192.168.4.2: bytes=32 time=10ms TTL=125
Reply from 192.168.4.2: bytes=32 time=10ms TTL=125
Reply from 192.168.4.2: bytes=32 time=13ms TTL=125

Ping statistics for 192.168.4.2:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 10ms, Maximum = 13ms, Average = 11ms

PC>ping 192.168.2.2

Pinging 192.168.2.2 with 32 bytes of data:

Reply from 192.168.2.2: bytes=32 time=10ms TTL=128
Reply from 192.168.2.2: bytes=32 time=0ms TTL=128
Reply from 192.168.2.2: bytes=32 time=13ms TTL=128
Reply from 192.168.2.2: bytes=32 time=9ms TTL=128

Ping statistics for 192.168.2.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 13ms, Average = 8ms
```

Figura 19. Prueba de conectividad de Alemania y Alemania 2

```
PC>ping 192.168.43.2

Pinging 192.168.43.2 with 32 bytes of data:

Request timed out.
Reply from 192.168.43.2: bytes=32 time=12ms TTL=124
Reply from 192.168.43.2: bytes=32 time=13ms TTL=124
Reply from 192.168.43.2: bytes=32 time=22ms TTL=124

Ping statistics for 192.168.43.2:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 12ms, Maximum = 22ms, Average = 15ms

PC>ping 192.168.80.2

Pinging 192.168.80.2 with 32 bytes of data:

Request timed out.
Reply from 192.168.80.2: bytes=32 time=13ms TTL=122
Reply from 192.168.80.2: bytes=32 time=14ms TTL=122
Reply from 192.168.80.2: bytes=32 time=17ms TTL=122

Ping statistics for 192.168.80.2:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 13ms, Maximum = 17ms, Average = 14ms
```

Figura 20.PING a España y Finlandia.

```
PC>ping 192.168.37.2

Pinging 192.168.37.2 with 32 bytes of data:

Request timed out.
Reply from 192.168.37.2: bytes=32 time=11ms TTL=125
Reply from 192.168.37.2: bytes=32 time=12ms TTL=125
Reply from 192.168.37.2: bytes=32 time=10ms TTL=125

Ping statistics for 192.168.37.2:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 10ms, Maximum = 12ms, Average = 11ms

PC>ping 192.168.44.2

Pinging 192.168.44.2 with 32 bytes of data:

Request timed out.
Reply from 192.168.44.2: bytes=32 time=0ms TTL=125
Reply from 192.168.44.2: bytes=32 time=14ms TTL=125
Reply from 192.168.44.2: bytes=32 time=12ms TTL=125

Ping statistics for 192.168.44.2:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 14ms, Average = 8ms
```

Figura 21.PING a Francia y Francia2

```
PC>ping 192.168.32.2

Pinging 192.168.32.2 with 32 bytes of data:

Request timed out.
Reply from 192.168.32.2: bytes=32 time=11ms TTL=124
Reply from 192.168.32.2: bytes=32 time=13ms TTL=124
Reply from 192.168.32.2: bytes=32 time=11ms TTL=124

Ping statistics for 192.168.32.2:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 11ms, Maximum = 13ms, Average = 11ms

PC>ping 192.168.19.2

Pinging 192.168.19.2 with 32 bytes of data:

Request timed out.
Reply from 192.168.19.2: bytes=32 time=12ms TTL=123
Reply from 192.168.19.2: bytes=32 time=22ms TTL=123
Reply from 192.168.19.2: bytes=32 time=11ms TTL=123

Ping statistics for 192.168.19.2:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 11ms, Maximum = 22ms, Average = 15ms
```

Figura 22. PING a Grecia y Croacia.

```
PC>ping 192.168.17.2

Pinging 192.168.17.2 with 32 bytes of data:

Request timed out.
Reply from 192.168.17.2: bytes=32 time=21ms TTL=124
Reply from 192.168.17.2: bytes=32 time=11ms TTL=124
Reply from 192.168.17.2: bytes=32 time=11ms TTL=124

Ping statistics for 192.168.17.2:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 11ms, Maximum = 21ms, Average = 14ms

PC>ping 192.168.35.2

Pinging 192.168.35.2 with 32 bytes of data:

Request timed out.
Reply from 192.168.35.2: bytes=32 time=7ms TTL=124
Reply from 192.168.35.2: bytes=32 time=11ms TTL=124
Reply from 192.168.35.2: bytes=32 time=12ms TTL=124

Ping statistics for 192.168.35.2:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 7ms, Maximum = 12ms, Average = 10ms
```

Figura 23.PING a Hungria2 e Irlanda.

```
PC>ping 192.168.8.2

Pinging 192.168.8.2 with 32 bytes of data:

Request timed out.
Reply from 192.168.8.2: bytes=32 time=0ms TTL=126
Reply from 192.168.8.2: bytes=32 time=9ms TTL=126
Reply from 192.168.8.2: bytes=32 time=0ms TTL=126

Ping statistics for 192.168.8.2:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 9ms, Average = 3ms

PC>ping 192.168.40.2

Pinging 192.168.40.2 with 32 bytes of data:

Request timed out.
Reply from 192.168.40.2: bytes=32 time=33ms TTL=124
Reply from 192.168.40.2: bytes=32 time=12ms TTL=124
Reply from 192.168.40.2: bytes=32 time=1ms TTL=124

Ping statistics for 192.168.40.2:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 1ms, Maximum = 33ms, Average = 15ms
```

Figura 24. PING a Israel e Islandia.

```
PC>ping 192.168.31.2

Pinging 192.168.31.2 with 32 bytes of data:

Request timed out.
Reply from 192.168.31.2: bytes=32 time=10ms TTL=125
Reply from 192.168.31.2: bytes=32 time=2ms TTL=125
Reply from 192.168.31.2: bytes=32 time=10ms TTL=125

Ping statistics for 192.168.31.2:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 2ms, Maximum = 10ms, Average = 7ms

PC>ping 192.168.73.2

Pinging 192.168.73.2 with 32 bytes of data:

Request timed out.
Reply from 192.168.73.2: bytes=32 time=25ms TTL=122
Reply from 192.168.73.2: bytes=32 time=16ms TTL=122
Reply from 192.168.73.2: bytes=32 time=14ms TTL=122

Ping statistics for 192.168.73.2:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 14ms, Maximum = 25ms, Average = 18ms
```

Figura 25.PING a Italia y Lituania.

```
PC>ping 192.168.6.2

Pinging 192.168.6.2 with 32 bytes of data:

Request timed out.
Reply from 192.168.6.2: bytes=32 time=24ms TTL=126
Reply from 192.168.6.2: bytes=32 time=13ms TTL=126
Reply from 192.168.6.2: bytes=32 time=12ms TTL=126

Ping statistics for 192.168.6.2:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 12ms, Maximum = 24ms, Average = 16ms

PC>ping 192.168.77.2

Pinging 192.168.77.2 with 32 bytes of data:

Request timed out.
Reply from 192.168.77.2: bytes=32 time=12ms TTL=123
Reply from 192.168.77.2: bytes=32 time=16ms TTL=123
Reply from 192.168.77.2: bytes=32 time=24ms TTL=123

Ping statistics for 192.168.77.2:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 12ms, Maximum = 24ms, Average = 17ms
```

Figura 26. PING a Luxemburgo y Letonia.

```
PC>ping 192.168.18.2

Pinging 192.168.18.2 with 32 bytes of data:

Request timed out.
Reply from 192.168.18.2: bytes=32 time=13ms TTL=123
Reply from 192.168.18.2: bytes=32 time=22ms TTL=123
Reply from 192.168.18.2: bytes=32 time=12ms TTL=123

Ping statistics for 192.168.18.2:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 12ms, Maximum = 22ms, Average = 15ms

PC>ping 192.168.53.2

Pinging 192.168.53.2 with 32 bytes of data:

Request timed out.
Reply from 192.168.53.2: bytes=32 time=13ms TTL=123
Reply from 192.168.53.2: bytes=32 time=23ms TTL=123
Reply from 192.168.53.2: bytes=32 time=26ms TTL=123

Ping statistics for 192.168.53.2:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 13ms, Maximum = 26ms, Average = 20ms
```

Figura 27.PING a Montenegro y Macedonia.

```
PC>ping 192.168.46.1

Pinging 192.168.46.1 with 32 bytes of data:

Reply from 192.168.46.1: bytes=32 time=0ms TTL=252
Reply from 192.168.46.1: bytes=32 time=7ms TTL=252
Reply from 192.168.46.1: bytes=32 time=12ms TTL=252
Reply from 192.168.46.1: bytes=32 time=19ms TTL=252

Ping statistics for 192.168.46.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 19ms, Average = 9ms

PC>ping 192.168.3.2

Pinging 192.168.3.2 with 32 bytes of data:

Request timed out.
Reply from 192.168.3.2: bytes=32 time=10ms TTL=126
Reply from 192.168.3.2: bytes=32 time=10ms TTL=126
Reply from 192.168.3.2: bytes=32 time=14ms TTL=126

Ping statistics for 192.168.3.2:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 10ms, Maximum = 14ms, Average = 11ms
```

Figura 28.PING a Malta y Holanda.

```
PC>ping 192.168.78.2

Pinging 192.168.78.2 with 32 bytes of data:

Request timed out.
Reply from 192.168.78.2: bytes=32 time=24ms TTL=124
Reply from 192.168.78.2: bytes=32 time=11ms TTL=124
Reply from 192.168.78.2: bytes=32 time=11ms TTL=124

Ping statistics for 192.168.78.2:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 11ms, Maximum = 24ms, Average = 15ms

PC>ping 192.168.5.2

Pinging 192.168.5.2 with 32 bytes of data:

Request timed out.
Reply from 192.168.5.2: bytes=32 time=13ms TTL=126
Reply from 192.168.5.2: bytes=32 time=11ms TTL=126
Reply from 192.168.5.2: bytes=32 time=11ms TTL=126

Ping statistics for 192.168.5.2:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 11ms, Maximum = 13ms, Average = 11ms
```

Figura29. PING Noruega y Polonia

```
PC>ping 192.168.38.2

Pinging 192.168.38.2 with 32 bytes of data:

Request timed out.
Reply from 192.168.38.2: bytes=32 time=15ms TTL=124
Reply from 192.168.38.2: bytes=32 time=22ms TTL=124
Reply from 192.168.38.2: bytes=32 time=25ms TTL=124

Ping statistics for 192.168.38.2:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 15ms, Maximum = 25ms, Average = 20ms

PC>ping 192.168.22.2

Pinging 192.168.22.2 with 32 bytes of data:

Request timed out.
Reply from 192.168.22.2: bytes=32 time=23ms TTL=123
Reply from 192.168.22.2: bytes=32 time=13ms TTL=123
Reply from 192.168.22.2: bytes=32 time=12ms TTL=123

Ping statistics for 192.168.22.2:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 12ms, Maximum = 23ms, Average = 16ms
```

Figura30.PING a Portugal y Rumania.

```
PC>ping 192.168.21.2

Pinging 192.168.21.2 with 32 bytes of data:

Request timed out.
Reply from 192.168.21.2: bytes=32 time=25ms TTL=123
Reply from 192.168.21.2: bytes=32 time=25ms TTL=123
Reply from 192.168.21.2: bytes=32 time=11ms TTL=123

Ping statistics for 192.168.21.2:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 11ms, Maximum = 25ms, Average = 20ms

PC>ping 192.168.74.2

Pinging 192.168.74.2 with 32 bytes of data:

Request timed out.
Reply from 192.168.74.2: bytes=32 time=11ms TTL=125
Reply from 192.168.74.2: bytes=32 time=11ms TTL=125
Reply from 192.168.74.2: bytes=32 time=20ms TTL=125

Ping statistics for 192.168.74.2:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 11ms, Maximum = 20ms, Average = 14ms
```

Figura 31.PING Serbia y Bielorrusia.

```
PC>ping 192.168.79.2

Pinging 192.168.79.2 with 32 bytes of data:

Request timed out.
Reply from 192.168.79.2: bytes=32 time=12ms TTL=123
Reply from 192.168.79.2: bytes=32 time=14ms TTL=123
Reply from 192.168.79.2: bytes=32 time=11ms TTL=123

Ping statistics for 192.168.79.2:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 11ms, Maximum = 14ms, Average = 12ms

PC>ping 192.168.33.2

Pinging 192.168.33.2 with 32 bytes of data:

Request timed out.
Reply from 192.168.33.2: bytes=32 time=143ms TTL=123
Reply from 192.168.33.2: bytes=32 time=10ms TTL=123
Reply from 192.168.33.2: bytes=32 time=6ms TTL=123

Ping statistics for 192.168.33.2:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 6ms, Maximum = 143ms, Average = 53ms
```

Figura 32 PING Suecia y Eslovenia.

```
PC>ping 192.168.67.2

Pinging 192.168.67.2 with 32 bytes of data:

Request timed out.
Reply from 192.168.67.2: bytes=32 time=4ms TTL=123
Reply from 192.168.67.2: bytes=32 time=4ms TTL=123
Reply from 192.168.67.2: bytes=32 time=14ms TTL=123

Ping statistics for 192.168.67.2:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 4ms, Maximum = 14ms, Average = 7ms

PC>ping 192.168.20.2

Pinging 192.168.20.2 with 32 bytes of data:

Request timed out.
Reply from 192.168.20.2: bytes=32 time=14ms TTL=123
Reply from 192.168.20.2: bytes=32 time=5ms TTL=123
Reply from 192.168.20.2: bytes=32 time=15ms TTL=123

Ping statistics for 192.168.20.2:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 5ms, Maximum = 15ms, Average = 11ms
```

Figura 33.PING a Moldavia y Eslovaquia.

```
PC>ping 192.168.9.2

Pinging 192.168.9.2 with 32 bytes of data:

Request timed out.
Reply from 192.168.9.2: bytes=32 time=0ms TTL=126
Reply from 192.168.9.2: bytes=32 time=22ms TTL=126
Reply from 192.168.9.2: bytes=32 time=0ms TTL=126

Ping statistics for 192.168.9.2:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 22ms, Average = 7ms

PC>ping 192.168.35.2

Pinging 192.168.35.2 with 32 bytes of data:

Reply from 192.168.35.2: bytes=32 time=2ms TTL=124
Reply from 192.168.35.2: bytes=32 time=0ms TTL=124
Reply from 192.168.35.2: bytes=32 time=1ms TTL=124
Reply from 192.168.35.2: bytes=32 time=0ms TTL=124

Ping statistics for 192.168.35.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 2ms, Average = 0ms
```

Figura 134. PING a Turquía y UK.

```
Packet Tracer PC Command Line 1.0
PC>ping 192.168.75.2

Pinging 192.168.75.2 with 32 bytes of data:

Request timed out.
Reply from 192.168.75.2: bytes=32 time=15ms TTL=125
Reply from 192.168.75.2: bytes=32 time=2ms TTL=125
Reply from 192.168.75.2: bytes=32 time=13ms TTL=125

Ping statistics for 192.168.75.2:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 2ms, Maximum = 15ms, Average = 10ms
```

Figura 35. PING a Ucrania.

Apéndice C: Prueba de conectividad usando OSPF v2 en emulación.

```
C:\Users\Alemania>ping 192.168.12.1

Haciendo ping a 192.168.12.1 con 32 bytes de datos:
Respuesta desde 192.168.12.1: bytes=32 tiempo=168ms TTL=253
Respuesta desde 192.168.12.1: bytes=32 tiempo=156ms TTL=253
Respuesta desde 192.168.12.1: bytes=32 tiempo=222ms TTL=253
Respuesta desde 192.168.12.1: bytes=32 tiempo=156ms TTL=253

Estadísticas de ping para 192.168.12.1:
    Paquetes: enviados = 4, recibidos = 4, perdidos = 0
    (0% perdidos),
Tiempos aproximados de ida y vuelta en milisegundos:
    Mínimo = 156ms, Máximo = 222ms, Media = 175ms

C:\Users\Alemania>ping 192.168.4.1

Haciendo ping a 192.168.4.1 con 32 bytes de datos:
Respuesta desde 192.168.4.1: bytes=32 tiempo=170ms TTL=255
Respuesta desde 192.168.4.1: bytes=32 tiempo=76ms TTL=255
Respuesta desde 192.168.4.1: bytes=32 tiempo=47ms TTL=255
Respuesta desde 192.168.4.1: bytes=32 tiempo=35ms TTL=255

Estadísticas de ping para 192.168.4.1:
    Paquetes: enviados = 4, recibidos = 4, perdidos = 0
    (0% perdidos),
Tiempos aproximados de ida y vuelta en milisegundos:
    Mínimo = 35ms, Máximo = 170ms, Media = 82ms

C:\Users\Alemania>ping 192.168.15.1

Haciendo ping a 192.168.15.1 con 32 bytes de datos:
Respuesta desde 192.168.15.1: bytes=32 tiempo=180ms TTL=253
Respuesta desde 192.168.15.1: bytes=32 tiempo=271ms TTL=253
Respuesta desde 192.168.15.1: bytes=32 tiempo=255ms TTL=253
Respuesta desde 192.168.15.1: bytes=32 tiempo=295ms TTL=253

Estadísticas de ping para 192.168.15.1:
    Paquetes: enviados = 4, recibidos = 4, perdidos = 0
    (0% perdidos),
Tiempos aproximados de ida y vuelta en milisegundos:
    Mínimo = 180ms, Máximo = 295ms, Media = 250ms

C:\Users\Alemania>
```

Figura 36.PING a Rep. Checa, Alemania y Dinamarca.

```
C:\Users\Alemania>ping 192.168.14.1

Haciendo ping a 192.168.14.1 con 32 bytes de datos:
Respuesta desde 192.168.14.1: bytes=32 tiempo=419ms TTL=253
Respuesta desde 192.168.14.1: bytes=32 tiempo=182ms TTL=253
Respuesta desde 192.168.14.1: bytes=32 tiempo=452ms TTL=253
Respuesta desde 192.168.14.1: bytes=32 tiempo=218ms TTL=253

Estadísticas de ping para 192.168.14.1:
    Paquetes: enviados = 4, recibidos = 4, perdidos = 0
    (0% perdidos),
Tiempos aproximados de ida y vuelta en milisegundos:
    Mínimo = 182ms, Máximo = 452ms, Media = 317ms

C:\Users\Alemania>ping 192.168.43.1

Haciendo ping a 192.168.43.1 con 32 bytes de datos:
Respuesta desde 192.168.43.1: bytes=32 tiempo=409ms TTL=252
Respuesta desde 192.168.43.1: bytes=32 tiempo=155ms TTL=252
Respuesta desde 192.168.43.1: bytes=32 tiempo=429ms TTL=252
Respuesta desde 192.168.43.1: bytes=32 tiempo=274ms TTL=252

Estadísticas de ping para 192.168.43.1:
    Paquetes: enviados = 4, recibidos = 4, perdidos = 0
    (0% perdidos),
Tiempos aproximados de ida y vuelta en milisegundos:
    Mínimo = 155ms, Máximo = 429ms, Media = 316ms

C:\Users\Alemania>ping 192.168.80.1

Haciendo ping a 192.168.80.1 con 32 bytes de datos:
Respuesta desde 192.168.80.1: bytes=32 tiempo=764ms TTL=250
Respuesta desde 192.168.80.1: bytes=32 tiempo=463ms TTL=250
Respuesta desde 192.168.80.1: bytes=32 tiempo=504ms TTL=250
Respuesta desde 192.168.80.1: bytes=32 tiempo=382ms TTL=250

Estadísticas de ping para 192.168.80.1:
    Paquetes: enviados = 4, recibidos = 4, perdidos = 0
    (0% perdidos),
Tiempos aproximados de ida y vuelta en milisegundos:
    Mínimo = 382ms, Máximo = 764ms, Media = 528ms
```

Figura 37. PING a Estonia, España y Finlandia.

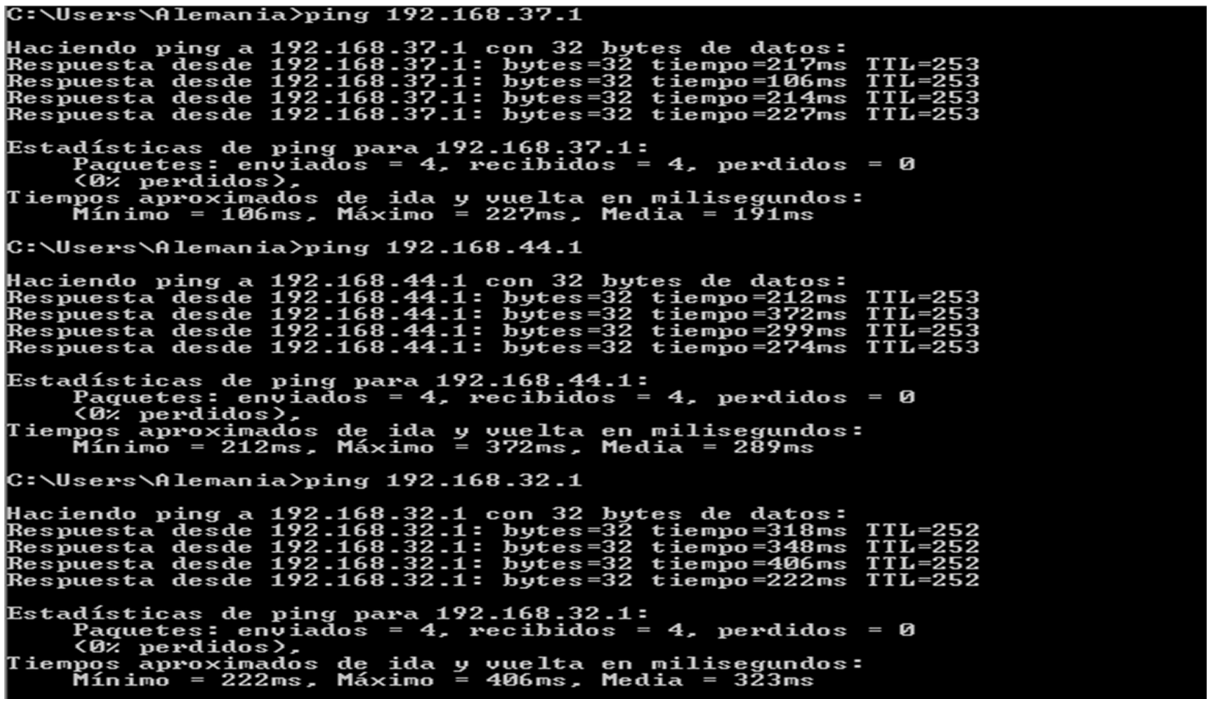


Figura 38. PING a Francia, Francia 2 y Grecia.

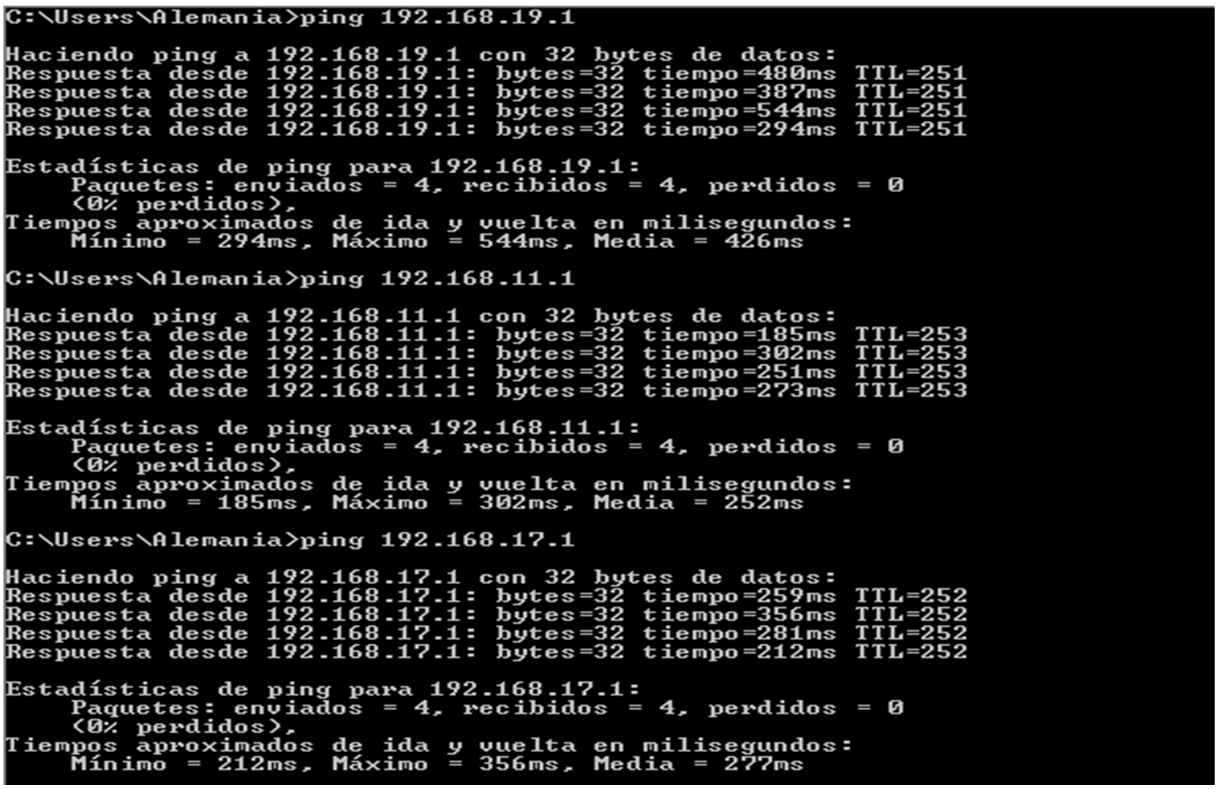


Figura 39. PING a Croacia, Hungría y Hungría 2.

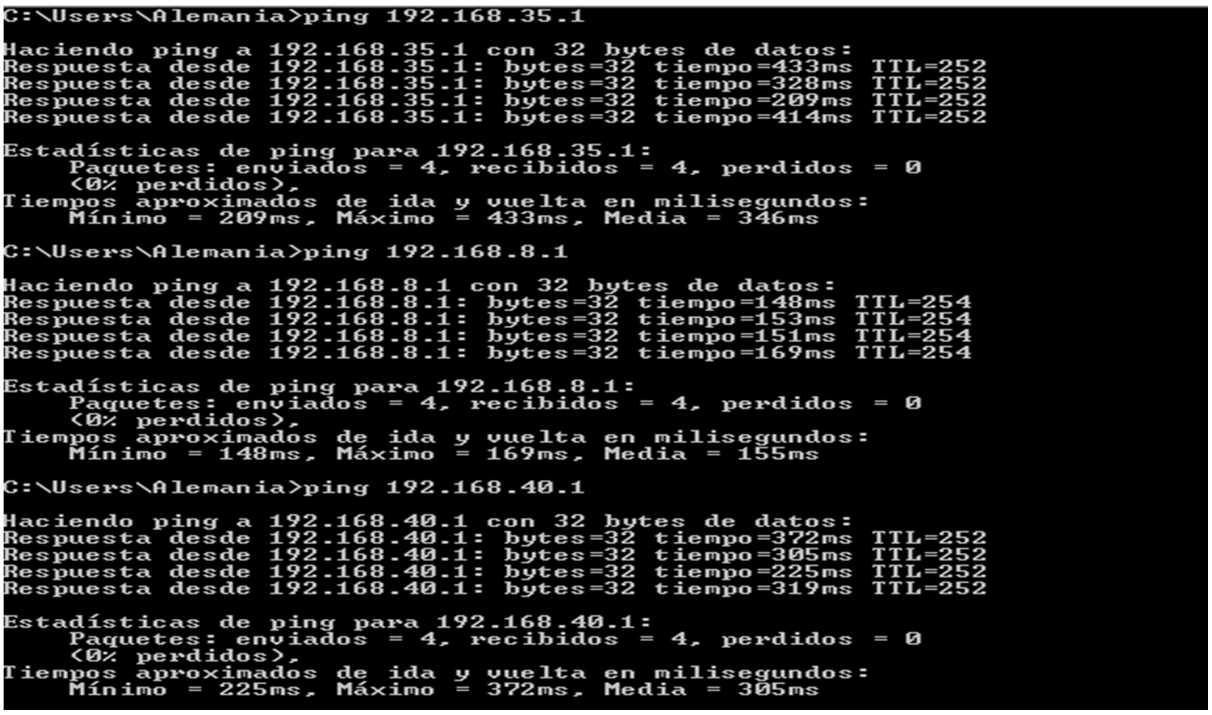


Figura 40. PING a Irlanda, Israel e Islandia.

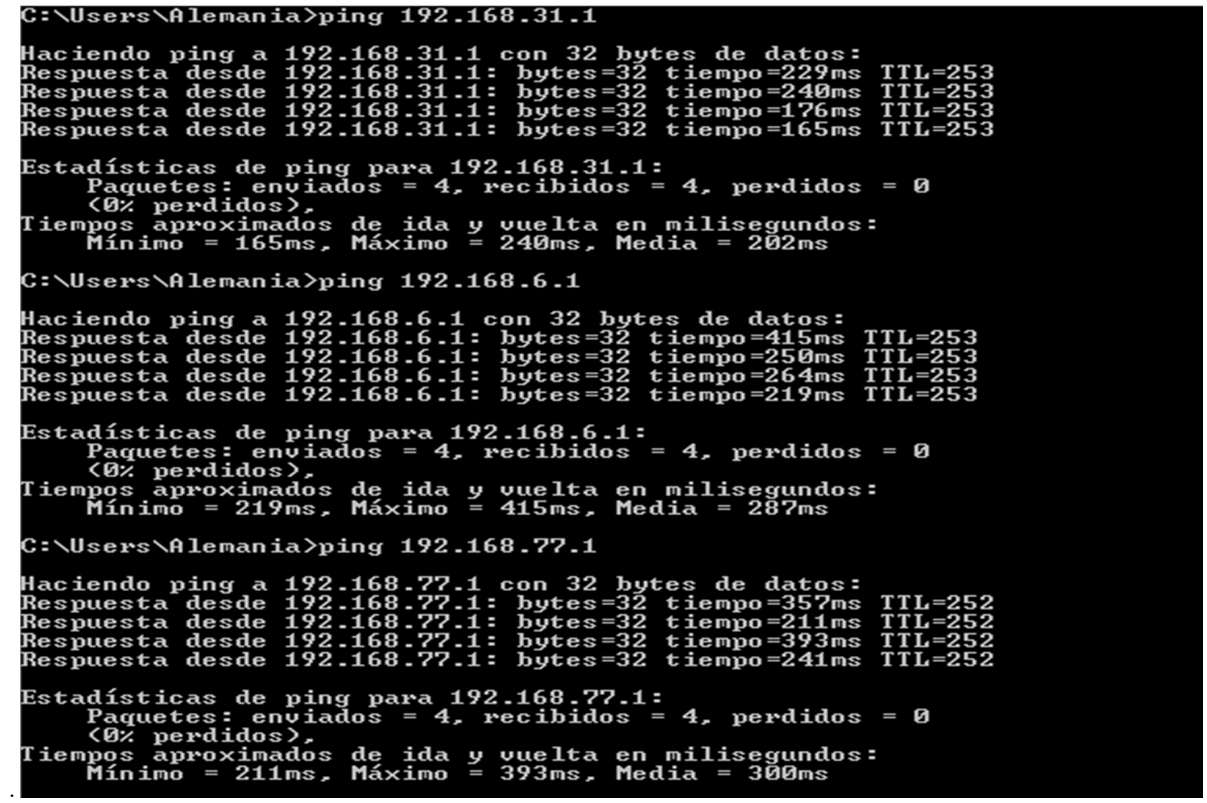


Figura 41. PING a Italia, Luxemburgo y Letonia

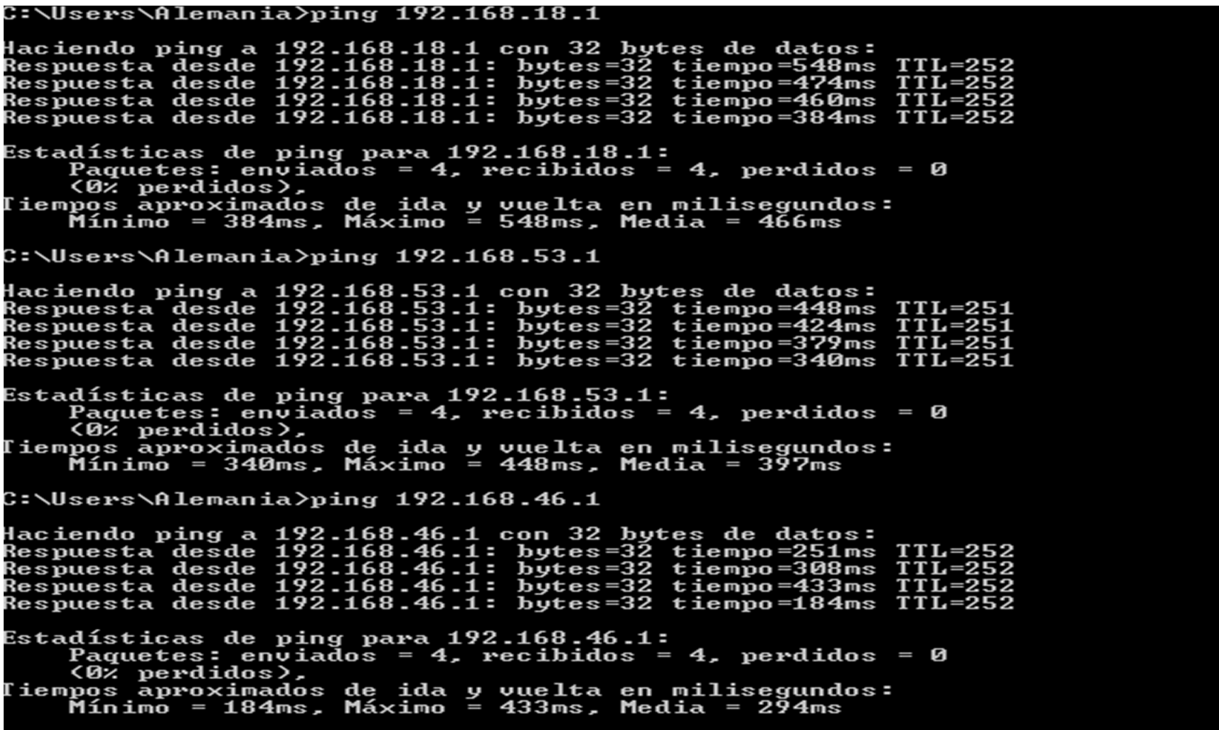


Figura 42. PING a Montenegro, Macedonia y Malta.

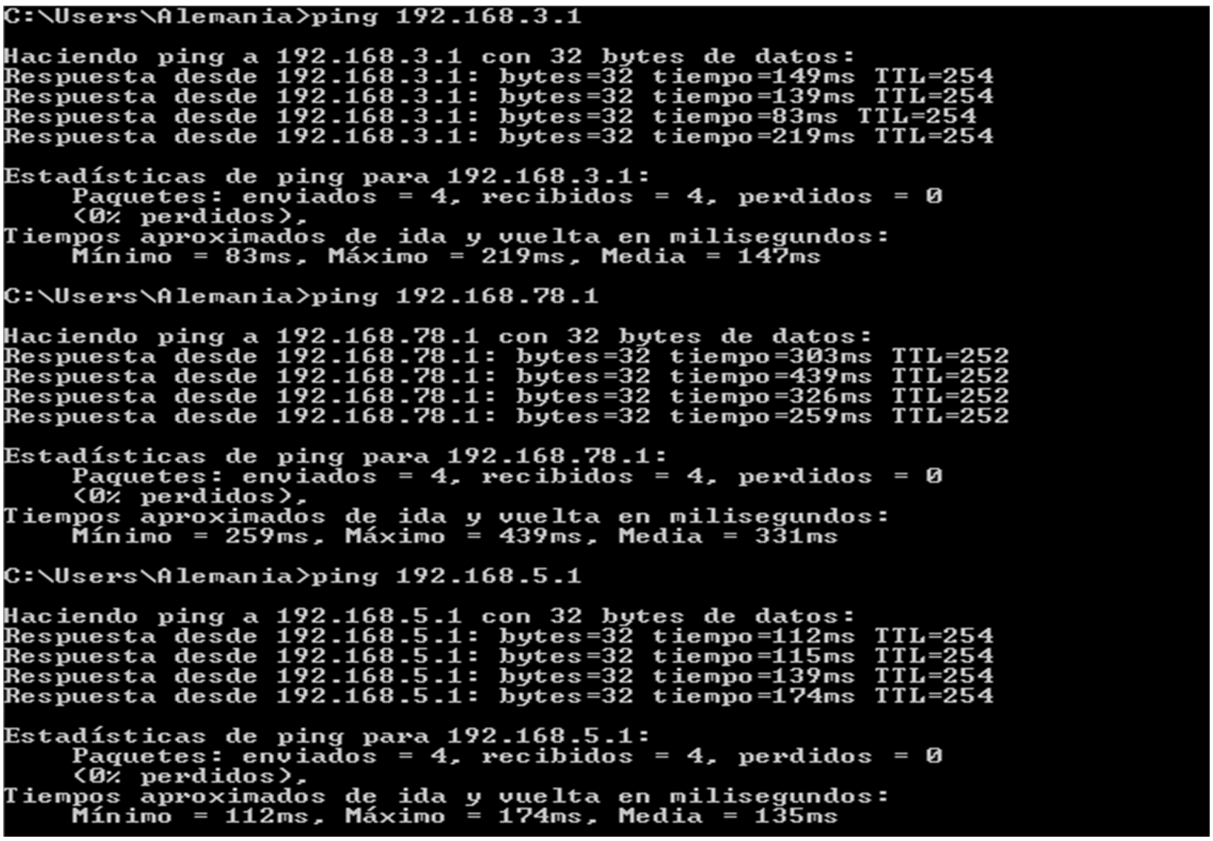


Figura 43. PING a Holanda, Noruega y Polonia.

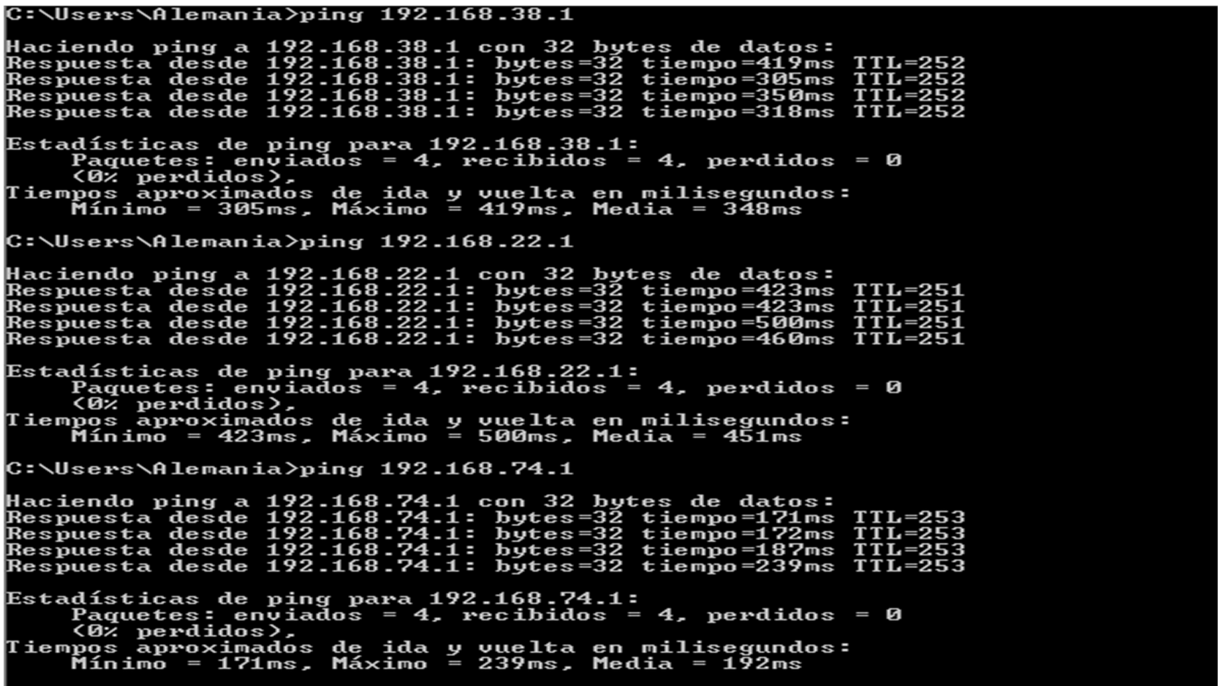


Figura 44. PING a Portugal, Noruega y Bielorrusia.

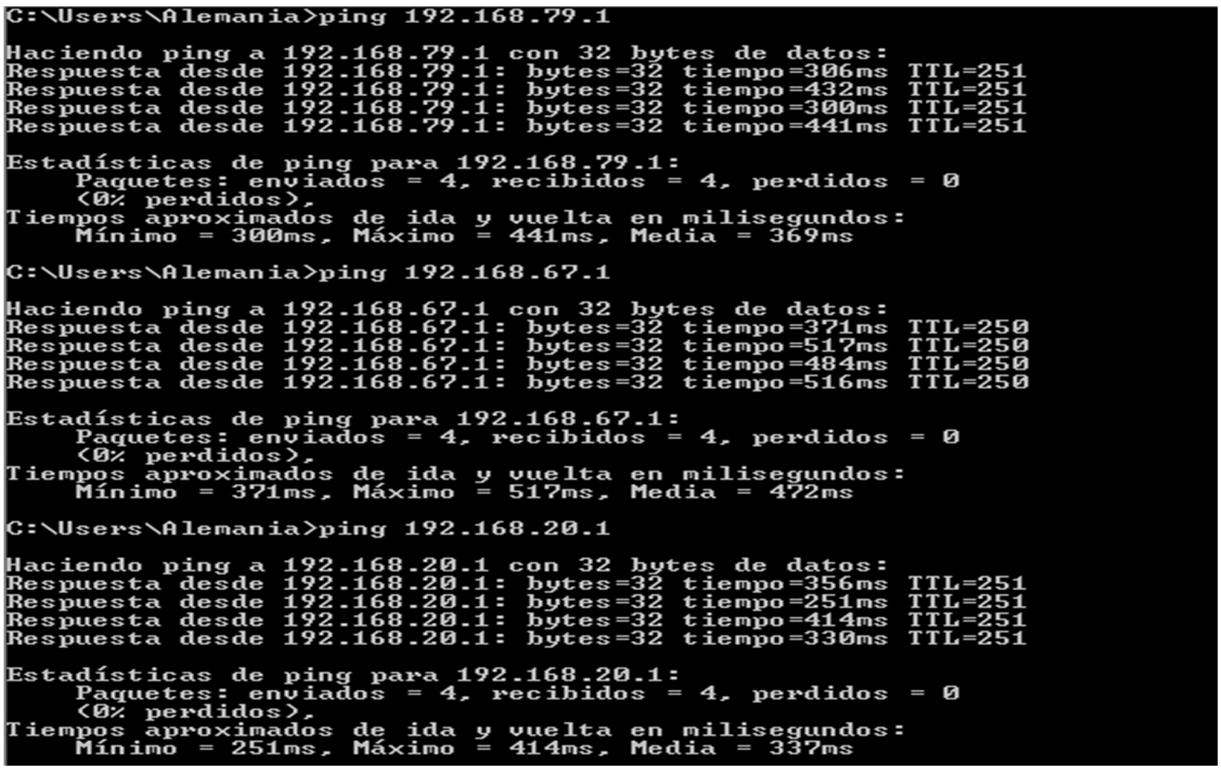


Figura 45. PING a Suecia, Moldavia y Eslovaquia.

Figura 46. PING a Turquía y Ucrania.

Referencias.

[1] Oliver Martin. *The "Hidden" Prehistory of European Research Networking*. EEUU: [en línea]; 2012: Trafford, ISBN: 978-1-4669-3872-4 disponible:
https://books.google.com.mx/books?id=eTRYAAAAQBAJ&pg=PT161&lpg=PT161&dq=sercnet+network++uk&source=bl&ots=UrRhMqpw22&sig=8YP9QomcBMO7Jcfl62L7JmFr6YU&hl=es&sa=X&ved=0ahUKEwjg5rvir7RAhViw1QKHcMGDjMQ6AEISDAG#v=onepage&q=sercnet%20network%20%20uk&f=false

[2] Castillo Velázquez José Ignacio. *Redes de datos, Contexto y evolución*, Segunda edición, México: 2016: SAMSARA, ISBN: 978-970-94-2968-8.

[3] Sanz Miguel A. *Fundamentos históricos de la Internet en Europa y en España*, SATEC S.A, [en línea]; 2007, disponible:
http://www.rediris.es/difusion/publicaciones/boletin/45/enfoque2.html

[4] Kaarina Lehtisalo. *The History of nordunet*, libro digital.

[5] Fluckiger Francois. *The European Researchers' Network Francois Fluckiger*, CERN, Geneva, [en línea] 2000, archivo digital; Disponible:
https://fluckiger.web.cern.ch/Fluckiger/Articles/F.Fluckiger-The_European_Researchers_Network.pdf

[6] SURFnet bv. *Surnet International*, the Annual Reports from 1988 to 2006, the SURFnet Bulletin periodical from 1987 to 2007, archivo digital.

[7] *The Hystory of the EARN Network*, Disponible:
http://earn-history.net/

[8] DANTE Archive. *Phare COSINE*; Disponible:
http://archive.dante.net/Backbones/Phare/Pages/Phare.aspx

[9]- Iesnews, *Esprit Information Exchange System. Welcome to Archive of European Integration*. Issue No. 11 August 1987, archivo digital.

[10] *The Europan Research Network 1989*: archivo digital.

[11] DANTE Archive ConneXions. *Dante and Europanet, #4.* The Interoperability Report, Vol 8 No.6, 1994. Archivo digital.

[12] Bersee Josefien, *DANTE and EuropaNet a profile* #4, Vol 8. No. 6 año 1994. Archivo Digital.

[13] Howard Davies, General Manager of DANTE, *Operational Network Services for the European Research Community*, in San Francisco, California, August 1993. Archivo digital.

[14] The Journal of Information Networking. *International network services in Europe and the role of DANTE*. Volume 1 Number 3, 1994. Archivo Digital.

[15] DANTE Archive. *20 years of DANTE;* Disponible:
https://dante.archive.geant.org/About_Us/20_years_of_DANTE/Pages/20_Years_of_Networking_Excellence.aspx

[16] DANTE Archive. *A blueprint for the next generation research network in Europe*, Eukarien. Archivo Digital.

[17] DANTE Archive. *European Cooperation for Academic and Industrial Research Networking (Eureka Project 1061)*; disponible:
http://archive.dante.net/Backbones/EuroCAIRN/Pages/EuroCAIRN.aspx
[18] DANTE Archive. *The EuroCAIRN Project, Archivo digital.*
[19] CERN. *World Wide Web: Global Networking*. Archivo Digital.
[20] *The birth of the web;* Disponible:
https://home.cern/topics/birth-web
[21] Barry M. Leiner, Vinton G. Cerf, David D. Clark, Robert E. Kahn, Leonard Kleinrock, Daniel C. Lynch, Jon Postel, Larry G. Roberts, Stephen Wolff. *Brief History of the Internet,* 1997, archive digital.
[22] DANTE Archive. *European Research Networking Moves into the Fast Lane*; disponible:
http://archive.dante.net/Backbones/TEN-34/Pages/TEN-34.aspx
[23] DANTE Archive. *Intercontinental Connectivity Developments as reported in The Works of DANTE 1993-1997*; Disponible:
https://new-archive.dante.net/Backbones/Documents/IC93-97-TWOD.pdf
[24] DANTE Archive. *TEN-34 Network Maps*, Archivo digital.
[25] DANTE Archive. *Quality Network Technology for User Oriented Multi Media*; Disponible:
http://archive.dante.net/Backbones/QUANTUM/Pages/QUANTUM.aspx
[26] Networks Research Laboratory, University of Cyprus. *Q-MED: Quality Network Technology for User-Oriented Multimedia in the Eastern Mediterranean Region*; disponible:
http://www.netrl.cs.ucy.ac.cy/index2.php?option=com_content&do_pdf=1&id=56
[27] Comisión Europea. *Sustitución de la TEN-34*, 1998. Archivo digital
[28] DANTE Archive. *TEN-155*; Disponible:
https://new-archive.dante.net/Backbones/TEN-155/Pages/TEN-155.aspx
[29] DANTE Archive. *European Research Networking Moves into the Fast Lane*; Disponible:
http://archive.dante.net/Backbones/Documents/TEN-155broch9810.pdf
[30] DANTE Archive. *Topology TEN-155 sep-1998*; Disponible:
https://new-archive.dante.net/Backbones/Documents/ten-155-1998-09.pdf
[31] DANTE Archive. *Topology TEN-155 may 2001*; disponible:
http://www.gateway.nameflow.net/ten-155/ten155net.gif
[32] GEANT.*GÉANT pan-European network;* Disponible:
https://www.geant.org/Networks/Pan-European_network/Pages/Home.aspx
[33] GEANT Project. *History of GEANT*; disponible:
http://geant3plus.archive.geant.net/About/Value_of_GEANT/Pages/History_of_GEANT.aspx
[34] *SurFnet international*, archivo digital.
[35] GEANT 3 Plus. *GEANT Topology 2001*, archivo digital.
[36] Comisión Europea, *La UE mejora la red paneuropea de investigación*; Disponible:
https://cordis.europa.eu/news/rcn/22563_es.html

[37] GEANT Archive. *Topology GEANT April 2004,* archivo digital.

[38] DANTE Archive. *El proyecto ALICE crea la primera red latinoamericana de investigación y educación*; Disponible:

http://alice1.archive.dante.net/server/show/conWebDoc.1257.html

[39] DANTE Archive. *El proyecto ALICE crea la primera red latinoamericana de investigación y educación*; disponible:

http://alice1.archive.dante.net/server/show/conWebDoc.1257.html

[40] UNAM, Sánchez Yllanez José Antonio. *América Latina y Europa unidas a través de Internet 2*, [En línea]: Año 4, Número 35, Enero de 2005; Disponible:

http://www.enterate.unam.mx/Articulos/2005/enero/unidas.htm

[41] DANTE Archive. *Topología del proyecto ALICE 2004*, archivo digital.

[42] DANTE Archive. *Topología red CLARA 2008*, archivo digital.

[43] GEANT Archive. *La nueva red paneuropea llega a España*, archivo digital.

[44] GEANT 2. *Topology GEANT 2*, Archivo digital.

[45] DANTE and GEANT 2. *GEANT 2 3rd Edition*, Archivo digital.

[46] GEANT 2. *Fogonadura*; disponible:

http://geant2.archive.geant.net/server/show/nav.00d009001.html

[47] Red CLARA, *Alice2, 2014;* Disponible:

https://www.redclara.net/index.php/conocimiento-e-innovacion/articulacion/finalizados/alice2

[48] Red CLARA, Topología de la red CLARA año 2014, Archivo Digital.

[49] TERENA, Bert van Pinxteren.*TERENA NREN Compendium,* 2010*, Archivo Digital.*

[50] GEANT 3. *Deliverable D5.29 (DS1.3.4, 2) GÉANT Service Uptake – Year 2*, archivo digital.

[51] GEANT Archive. *Geant 3*; Disponible:

https://geant3.archive.geant.org/Network/pages/home.aspx

[52] European commission. *Advancing research collaboration with GEANT´s high speed infrastructure*; Disponible:

https://ec.europa.eu/digital-single-market/en/news/advancing-research-collaboration-g%C3%A9ants-high-speed-infrastructure

[53] TERENA, Natalie Allred, Bert van Pinxteren. *GÉANT COMPENDIUM, of National Research and Education Networks in Europe 2015 Edition*, archivo digital.

[54] TERENA, *GÉANT COMPENDIUM, of National Research and Education Networks in Europe 2016 Edition*, Archivo Digital.

[55] Europan Union and GEANT, *GN3plus Innovation Programme Highlights*, archivo digital.

[56] Europan Union and GEANT, *GN3plus Year 2 Project Achievements,* Archivo Digital.

[57] GEANT. *GÉANT Project partners*; Disponible:

http://www.geant.org/Projects/GEANT_Project_GN4/Pages/Partners.aspx

[58] GEANT, Topology GEANT 2017. Archivo Digital.

[59] NORDUNET. *GEANT, Nordunet*; disponible:
https://www.nordu.net/content/g%C3%A9ant

[60] GEANT. *GÉANT pan-European network Europe's essential terabit-ready network is the most advanced and well-connected research and education network in the world*; Disponible:
https://www.geant.org/Networks/Pan-European_network/Pages/Home.aspx

[61] INFINERA. *DANTE and Infinera Deliver 2 Tb/s Capacity in Less Than 12 Minutes on GÉANT Production Network from Amsterdam to Frankfurt;* Disponible:
https://www.infinera.com/dante-and-infinera-deliver-2-tbs-capacity-in-less-than-12minutes-on-geant-production-network-from-amsterdam-to-frankfurt/

[62] Internet Society. *How it Works, Internet Society*; Disponible:
https://www.internetsociety.org/internet/how-it-works/

[63] R.F.C. 791, *Internet Protocol*

[64] GEANT 3 Archive. *OTN and NG-OTN: Overview*, Archivo digital.

[65] Cisco Network Academy, *Fundamentos de enrutamiento y conmutación (Routing and Switching Essentials),* CCNA Versión 5, compilado por Nicolás contador. Archivo Digital.

[66] Castillo Velásquez José Ignacio, *Switching and Routing introducción*, México, Samsara Editorial, 2016, ISBN: 978-970-94-2977-0

[67] DARPA, RFC 1058, *Routing Information Protocol,* 1988.

[68] Cisco System, Jeff Doyle, Jennifer Carroll. *Routing TCP/IP, Volume 1,* segunda edición, 2006. ISBN: 1-58705-202-4; disponible:
https://books.google.com.mx/books?id=JjdF2yWqJAwC&pg=PA169&lpg=PA169&dq=Hedrick+protocol+rip&source=bl&ots=Cf_Ardbyza&sig=CzLGgh_F87O8NyFRKEKtR2k267g&hl=es-419&sa=X&ved=0ahUKEwj9wNe9w_nQAhUEhlQKHZP8AYoQ6AEIPjAE#v=onepage&q=Hedrick%20protocol%20rip&f=false

[69] Sportack Mark A. Routing Fundamentals, Routing Information Protocol. Cisco Press, 1999; disponible:
https://www.cisco.com/cpress/cc/td/cpress/fund/ith2nd/it2444.htm

[70] Cisco Networking Academy, Jean Polo Cequeda Olago. *Conceptos y protocolos de enrutamiento*. Capítulo 1, Archivo digital.

[71] DARPA, RFC 2453, *RIP Versión 2, 1998.*

[72] Network Working Group. RFC 2080, RIPng for IPv6, 1997.

[73] DARPA, RFC 2328, *OSPF Version 2,* 1998.

[74] Cisco System, Jeff Doyle, *Routing TCP/IP, Volume 2,* segunda edición, 2017, Archivo digital

[75] Internet Society. *Open Shortest Path First: The State of The Link* State; disponible:
https://www.internetsociety.org/publications/ietf-journal-july-2015/open-shortest-path-first

[76] *DARPA, RFC 1247 OSPF Version 2 Tools*, 1991

[77] CISCO.*OSPF Neighbor States* Cisco; Disponible:
https://www.cisco.com/c/en/us/support/docs/ip/open-shortest-path-first-ospf/13685-13.html

[78] CISCO: *Configuring OSPFv2*, Archivo Digital; Disponible:
https://www.cisco.com/c/en/us/td/docs/switches/datacenter/sw/5_x/nx-os/unicast/configuration/guide/l3_cli_nxos/l3_ospf.pdf

[79] Chapman and Hall/CRC Computer and Information Science Series.*The Practical Handbook of Internet Computing*, Munindar P. Singh. ISBN: 1-58488-381-2; Disponible:
https://books.google.com.mx/books?id=XKF2gg9Wq08C&pg=SA52-PA6&lpg=SA52-PA6&dq=snmpv3+rfc&source=bl&ots=v2FxnX0TJF&sig=b4vOMPmNd5y6kppgqyo_8FEq2Qc&hl=es-419&sa=X&ved=0ahUKEwjowM6Yue7UAhVIPCYKHTLtAmQQ6AEIfDAJ#v=onepage&q=snmpv3%20rfc&f=false

[80] Network Working Group. R.F.C. 1157. *A Simple Network Management Protocol (SNMP) ,*1990.

[81] Charles M. Kozierok. *Book, TCP/ IP GUIDE,* EEUU, 2005, ISBN: 1-59327-047-X; Disponible:
http://index-of.es/Magazines/hakin9/books/No.Starch.TCP.IP.Guide.Oct.2005.pdf

[82] IEEE: NoticIEEEro, José Ignacio Castillo Velázquez. *El Árbol de internet y la estructura de la información de gestión de una red*, April 2009, Year 20, Number 62; Disponible:
https://issuu.com/noticieeero/docs/noticieeero_62/25

[83] IBM Corporation. *Simple Network Management Protocol (SNMP) Support,* 1997, Archivo digital.

[84] EARLANG, Ericsson AB. *Simple Network Management Protocol (SNMP),* 2017.Archivo Digital.

[85] R.F.C. 1901, *Introduction to Community-based SNMPv2*

[86] R.F.C.1905, *Protocol Operations for version 2 of the Simple Network Management Protocol (SNMPv2)*

[87] IBM Corporation. *Protocol data units (PDUs);* Disponible:
https://www.ibm.com/support/knowledgecenter/en/SSB23S_1.1.0.12/gtpc1/pdus.html

[88] IBM, *SNMP V3;* Disponible:
https://www.ibm.com/support/knowledgecenter/es/ssw_aix_61/com.ibm.aix.networkcomm/snmpv3_intro.htm

[89] R.F.C. 3414, *User-based Security Model (USM) for version 3 of the Simple Network Management Protocol (SNMPv3)*

[90] R.F.C 3412, *Message Processing and Dispatching for the Simple Network Management Protocol (SNMP)*