

Universidad Autónoma Metropolitana  
Unidad Azcapotzalco

**Departamento:** Ciencias Básicas e Ingeniería

**Licenciatura:** Ingeniería en Computación

Reporte.

Diseño e implementación de una aplicación  
para la Administración de una Red en 1.0 Linux

**Trimestre:** 10P

**Unidad Enseñanza Aprendizaje:** 115127

Proyecto Terminal de Ingeniería en  
Computación 2

**Profesor:** Francisco Javier Zaragoza Martínez

**Asesor:** Arturo Zúñiga López

**Alumno:** Héctor Ramiro Pérez Sánchez

**Matricula:** 203307712

# Tabla de contenidos

## Contenido

Tabla de contenidos .....	2
Resumen .....	3
Introducción.....	4
Gestión de la seguridad en una red computacional. ....	4
Justificación.....	4
Antecedentes.....	4
Objetivos.....	5
Metodología y desarrollo.....	6
Resultados y discusión .....	7
Conclusiones.....	8
Bibliografía.....	9
Apéndices .....	10
Manual de configuración del entorno.....	10
Casos de uso .....	45
Código fuente .....	82
Paquete GUI .....	82
Clase LoginGUI.....	82
Clase MainGUI.....	91
Paquete SNMPlogic.....	165
Clase Main.....	165
Clase ShowRunParser .....	166
Paquete SNMPlogic.almacenamiento .....	170
Clase Data.....	170
Paquete SNMPlogic.interfaces .....	175
Clase Protocolo.....	175
Paquete SNMPlogic.protocolos.....	176
Clase Telnet.....	176
Paquete SNMPlogic.tareas .....	184
Clase ShowRunInfo.....	184
Pruebas .....	194

## Resumen

Para este proyecto se pretendió realizar una aplicación computacional que gestionara la seguridad de una red computacional mediante un dispositivo Router. El alcance en el manejo de la seguridad del dispositivo Router es el básico limitado por las 300 horas de trabajo para este proyecto terminal. Se utiliza una metodología de desarrollo en espiral y programación orientada a objetos.

El resultado fue satisfactorio ya que la aplicación cubre todos los requerimientos básicos de seguridad del Router y de la red. También se procuró que la aplicación obedeciera a las buenas prácticas de configuración en los dispositivos Router.

El principal problema con el que se encontró fue la falta inicial de especificaciones de este proyecto, forzando a reconsiderar nuevas opciones constantemente y a cambiar de tecnologías, ya que éstas no se adecuaban a las necesidades del proyecto, en un par de ocasiones. Asimismo la falta de un alcance más específico provocó un divago en la línea de trabajo.

El resultado de este proyecto es una aplicación que satisfactoriamente cumple con los requerimientos básicos de seguridad del dispositivo Router y de la red que maneja. La aplicación realiza la gestión del dispositivo de una manera intuitiva para personas que tienen los conocimientos básicos de la gestión de redes en dispositivos Router de Cisco.

## **Introducción**

### ***Gestión de la seguridad en una red computacional.***

Cisco define la gestión de la seguridad como el control del acceso a los recursos de la red mediante técnicas de autenticación y políticas de autorización<sup>1</sup>.

### ***Justificación.***

En la actualidad las redes de comunicación juegan un papel preponderante en casi todos los sistemas productivos de la sociedad, especialmente en aquellos que demandan de una gran eficiencia en el almacenamiento, transmisión y acceso seguro de la información. En la actualidad para estas funciones se utilizan las computadoras, pero una computadora por sí sola es incapaz de realizar todas estas funciones y de atender a todos los usuarios, empresas, etc. que tienen estas necesidades. Para satisfacer esta demanda son necesarios una gran cantidad de dispositivos de diferentes tipos y con diferentes fines. Los simples dispositivos tampoco son capaces de realizar todas estas tareas si no se comunican entre sí y para ello es necesario una o varias redes que los intercomunicuen.

Las redes de computadoras pronto se volvieron complejas y se necesitó de dispositivos especiales para su administración. Estas redes pronto se vieron afectadas por el mal uso de los recursos disponibles en ellas, tales como información sensible, gasto irracional de recursos, etc. presentándose la necesidad de limitar o denegar el acceso a estos recursos reflejándose en gasto de tiempo, dinero, etc. Así surgió en la necesidad de generar tanto herramientas físicas (hardware) como herramientas computacionales (software) para satisfacer esta demanda de aseguramiento de los recursos de una red.

## **Antecedentes**

Uno de los más recientes y destacados productos computacionales (software) para realizar tareas de manejo en la seguridad de una red es el Cisco Router and Security Device Manager (SDM) el cual es una herramienta basada en Web que se encarga de la gestión de dispositivos Router de Cisco que aumenta la productividad en la gestión de las redes, simplifica los cambios a los Router y ayuda a encontrar los problemas en redes complejas. Este software representa un hito en la administración de las redes computacionales.

Este proyecto se basó en gran parte en un intento de proveer algunas de las funciones de seguridad del software previamente mencionado. Se optó por el desarrollo de la parte del manejo de la seguridad del Router debido a la carencia de productos similares a este que no fueran privativos.

---

<sup>1</sup> MIB Quick User Referente © página 24 segunda viñeta.

## **Objetivos**

Diseñar, desarrollar e implementar una aplicación para la gestión de seguridad en una red, cubriendo los principios básicos de seguridad en la red y en el dispositivo definidos en el libro CCNA Security Oficial Examination Guide© de Michael Waltkins y Kevin Wallace capítulos 1 y 5 para el dispositivo Router Cisco modelo 1700 con IOS de Cisco del tipo Security mediante una aplicación en la plataforma Java con él JDK (Java development kit) Version 6 Update 21 y para usarse con él JRE (Java Runtime Enviroment Version 6).

## Metodología y desarrollo

Se optó por un desarrollo en espiral ya que éste reduce los riesgos del proyecto, facilita el desarrollo de productos con calidad y de fácil mantenimiento. La selección del dispositivo se debe a la existencia de estos para el desarrollo de prácticas en la Universidad Autónoma Metropolitana (UAM), lo cual facilitó enormemente su utilización. Por otro lado la referencia bibliográfica principal antes mencionada se considera la más adecuada para este tema en particular. Se tuvo que discernir los temas y funcionalidades más comunes y útiles en la empresa mexicana hoy en día.

El desarrollo en la plataforma Java fue debido a tanto su factibilidad de trabajo multiplataforma como al hecho de que su IDE (ambiente de desarrollo integrado) de desarrollo, en este caso Netbeans es de libre distribución. Asimismo en Java se cuenta con una extensa cantidad de API's (interfase de programación aplicada) de libre distribución y con una extensa documentación. Otra razón para escoger Java fue la familiaridad con la tecnología SWING que permite realizar interfaces de usuario de una manera relativamente sencilla. Para establecer una conexión con el Router se utilizó una librería ya existente de apache disponible en Internet<sup>2</sup> con el nombre de "commons-net-current.jar" que provee todas las funciones de conexión con el Router a través del protocolo TELNET (terminal remota).

El desarrollo de esta aplicación se realizó en los siguientes pasos principales:

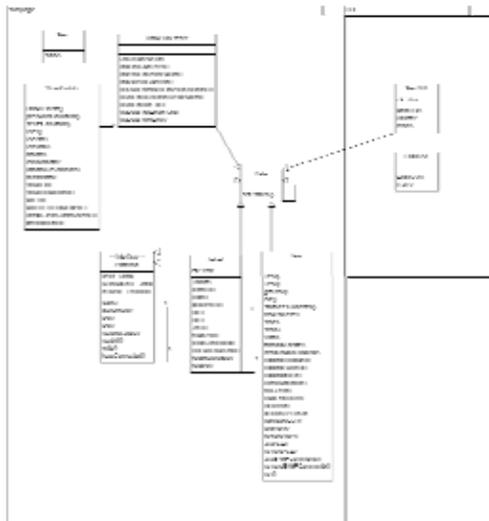
Investigación y adaptación a la tecnología usar.

Generación de casos de uso para definición de la funcionalidad.

Desarrollo de la aplicación.

Realización de pruebas.

La aplicación se basó en el siguiente diagrama de clases:



Nota: Para más detalle ir a los anexos.

<sup>2</sup> <http://archive.apache.org/dist/commons/net/>

## Resultados y discusión

El producto final se presenta como una aplicación Stand Alone que no requiere de ningún otro programa o servicio salvo de la instalación de la máquina virtual de Java. Esto ahorra tiempos evitando configuraciones extensas, promueve la movilidad de la aplicación y facilita su uso.

La aplicación consta principalmente de una interfaz de autenticación y una interfaz de configuración principal con 15 pestañas con funcionalidades individuales dentro de ellas. Cada pestaña gestiona un tipo de configuración en particular del Router. Se buscó un punto medio entre la funcionalidad, utilidad y buenas prácticas en la administración del equipo Router. Esto es, que a pesar de que algunas funcionalidades, por ejemplo borrar la contraseña de una línea, son posibles no se implementaron debido a que no obedece las buenas prácticas de administración de dispositivos de red. Algunas otras funcionalidades, si bien no siguen estrictamente las buenas prácticas, se implementaron debido a que a final de cuentas es la responsabilidad del administrador llevarlas a cabo ya que en casos particulares las buenas para prácticas de administración no son las más adecuadas para cubrir satisfactoriamente las necesidades del administrador.

Se procuró que la interfaz fuera intuitiva, pero no hasta grado de usuario final, ya que al final de cuentas esta es una aplicación para administradores de redes lo cual conlleva que el usuario debiera tener los conocimientos básicos de redes necesarios para entenderlo. Y debiera de tener el conocimiento de las consecuencias de las acciones que realiza sobre un dispositivo sensible como el Router. Por eso en cuanto al diseño de la interfaz se procuró hacer la minimalista y concisa. Tomando en cuenta la resolución mínima de  $800 \times 600$  que un monitor actual pudiera tener.

Salvó un par de excepciones la aplicación funciona mediante una serie de tareas, las cuales son provistas por las diferentes pestañas en la interfaz principal. Las tareas se agregan a una lista de tareas por ejecutarse que es visible en la parte inferior de la aplicación. Al presionar el botón de “añadir” en las pestañas se agregan tareas a las lista, hasta este momento ninguna instrucción se le ha enviado al Router, en cuanto se presiona el botón de ejecutar; todas las tareas en la lista se ejecutan y se guarda la nueva configuración en la NVRAM (la memoria de acceso aleatorio no volátil) del Router. Una vez hecho esto la nueva información del Router se carga en la interfaz principal. Las excepciones a esta situación son las pestañas de información general y auto asegurar que por su naturaleza se ejecutan directamente sobre la terminal del Router. La otra excepción de la pestaña de mensajes, ya que se requiere que ésta presente los mensajes en tiempo real flexible<sup>3</sup>.

---

<sup>3</sup> Se denominan las aplicaciones de tiempo real flexible como aquellas en las que un pequeño retraso es aceptable. La tecnología Java es incapaz de manejar aplicaciones a tiempo real inflexible, estas últimas se denominan como aquellas aplicaciones a las cuales un pequeño retraso en cuanto a su ejecución es inaceptable. La razón por la que Java es incapaz de manejar aplicaciones de tiempo real inflexible es que carece de destructores los cuales puedan liberar memoria en tiempo real. En vez de ellos tiene el sistema GARBAGE COLECTOR el cual de acuerdo a su algoritmo programado cuando se llenan la memoria de la máquina virtual, el recolector de basura congela la memoria RAM y la reordena. Esto conlleva a que la aplicación se congele algunos segundos de manera inesperada.

## **Conclusiones**

La aplicación desarrollada facilita en gran medida la gestión de los requerimientos básicos de seguridad del equipo y de la red ya que provee una interfaz gráfica relativamente amigable a comparación de la línea de comandos del sistema operativo del dispositivo Router. Proveyendo un entendimiento más fácil, directo y conciso de la situación de la configuración interna del Router, facilitando la comprensión en los cambios realizados, simplificando las tareas monótonas de autenticación y abreviando la gran cantidad de comandos de texto que se deben utilizar para configurar un dispositivo Router.

Esta aplicación podría ser utilizada en el entorno profesional, aunque muy limitada a comparación de otros productos de software privativo existentes en el mercado, en el entorno educacional, ésta aplicación es idónea como apoyo para cursos básicos de administración de redes.

Con los recursos necesarios esta aplicación puede expandirse hasta cubrir todas las necesidades de seguridad existentes, sólo se requiere un mayor detalle en el conocimiento de los sistemas operativos de los Router Cisco, así como de un contexto más real sobre las necesidades reales de seguridad de un sistema ya en producción.

## Bibliografía

Michael Walktins y Kevin Wallace “CCNA Security Oficial Examination Guide” primera edición junio de 2008 Cisco Press ISBN-10: 1-58720-220-4

Todd Lammle “CCNA: Cisco Certified Network Associate Study Guide: Exam 640-802, 6th Edition ISBN: 978-0-470-11008-9”

Información de cisco SDM <http://www.cisco.com/en/US/products/sw/secursw/ps5318/> con fecha .26/05/2010

Referencia de comandos básicos del Router [http://www.cisco.com/en/US/docs/ios/12\\_2/configfun/configuration/guide/fcf001.html](http://www.cisco.com/en/US/docs/ios/12_2/configfun/configuration/guide/fcf001.html) con fecha 26/05/2010

Configuración básica del equipo <http://es.kioskea.net/faq/2759-Router-cisco-configuracion-basica> con fecha 26/05/2010

Cisco AS5300 Universal Access Server Software Configuration Guide descargada de <http://www.cisco.com> con fecha 26/05/2010

Basic Router Configuration using SDM [http://www.cisco.com/en/US/products/sw/secursw/ps5318/products\\_configuration\\_example09186a008073e067.shtml](http://www.cisco.com/en/US/products/sw/secursw/ps5318/products_configuration_example09186a008073e067.shtml) con fecha 25/05/2010

Cisco configuration Professional for Cisco Access Routers ©Cisco Systems, Inc. <http://www.cisco.com> con fecha 26/05/2010

# Apéndices

## ***Manual de configuración del entorno.***

Manual de preparación de entorno de pruebas.

Se generó este manual debido a la inexistencia de un solo manual que abarque todos los elementos necesarios para su correcto funcionamiento.

Precondiciones: Se tiene instalado el programa GNS3 y se tiene corriendo una imagen de un Router (ruteador) cisco 1700 y un “idle pc” (porcentaje máximo del procesamiento a usar) adecuado a nuestro equipo. Ver guía GNS3-0.5-“tutorial.pdf” que se puede obtener de la siguiente página: <http://www.gns3.net/documentation>

Finalidad: Conectar la PC mediante un Ethernet virtual a la emulación del Router (ruteador) Cisco corriendo en el programa de GNS3

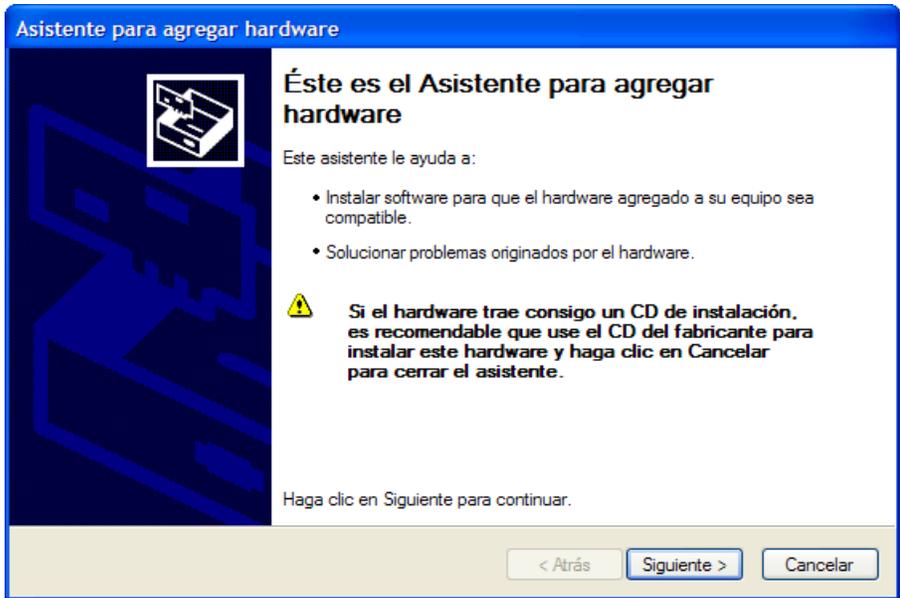
Parte 1

Finalidad: Creación de un Adaptador de bucle invertido.

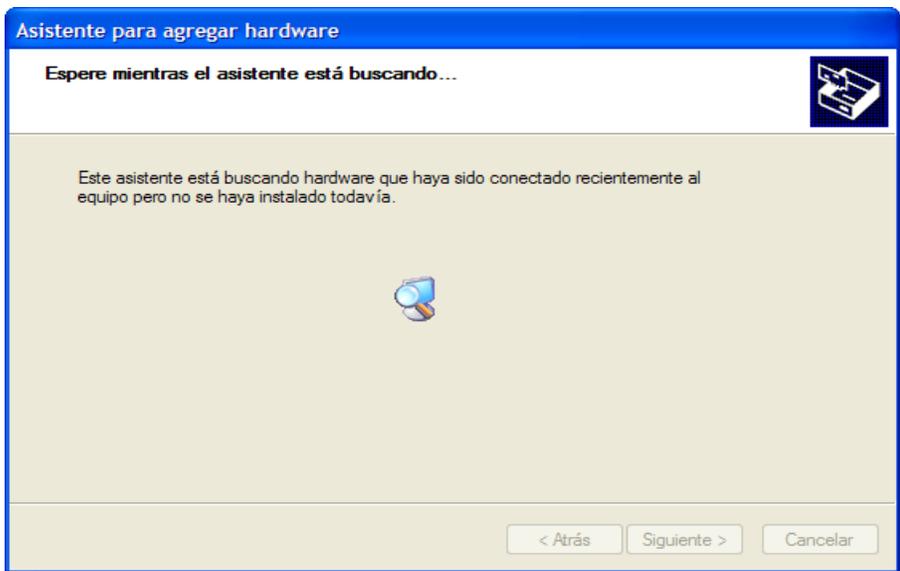
*Inicio --> Panel de control*



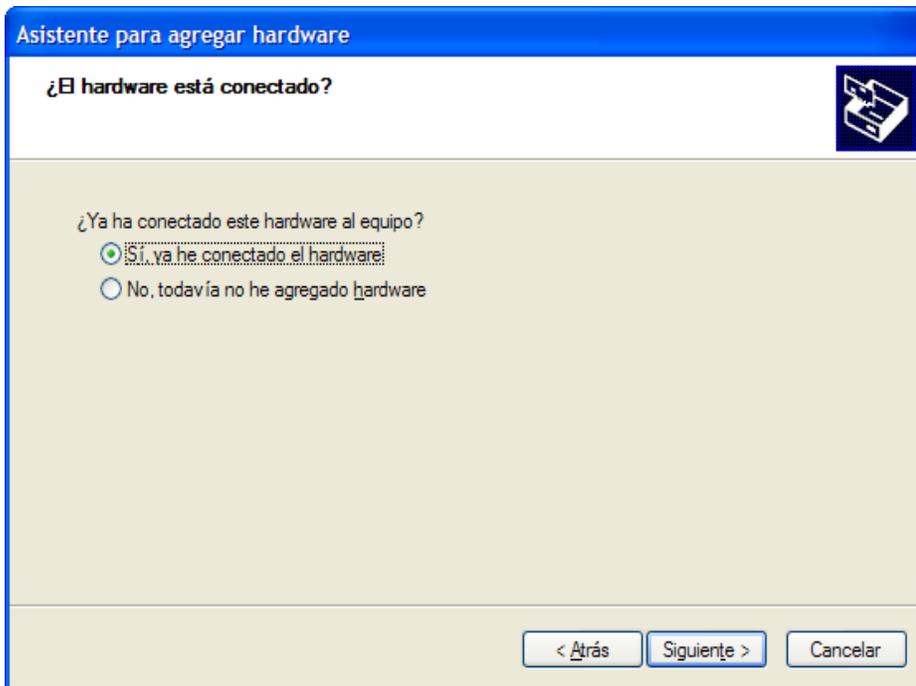
*Agregar hardware*



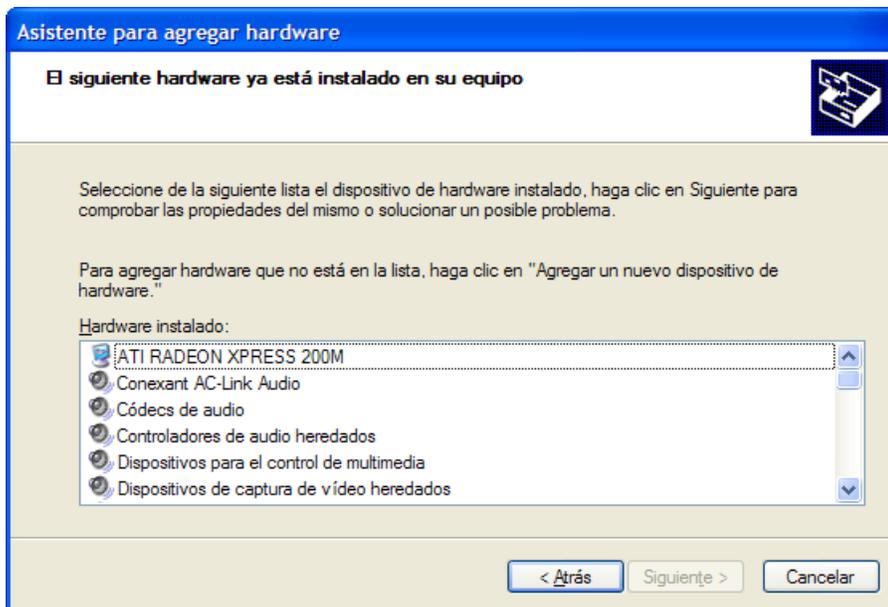
Siguiente tardará unos segundos en intentar encontrar un hardware nuevo.



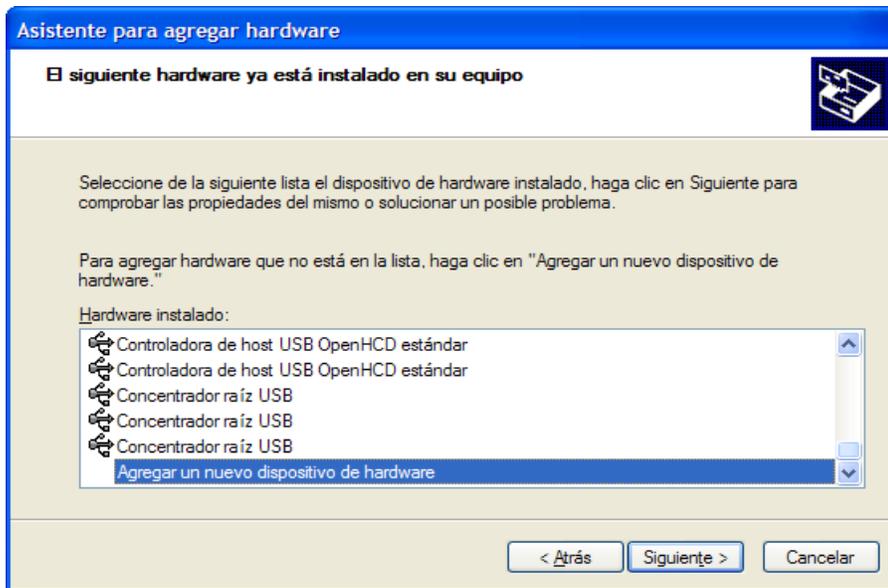
Como no hemos conectado nada, no encuentra nada.



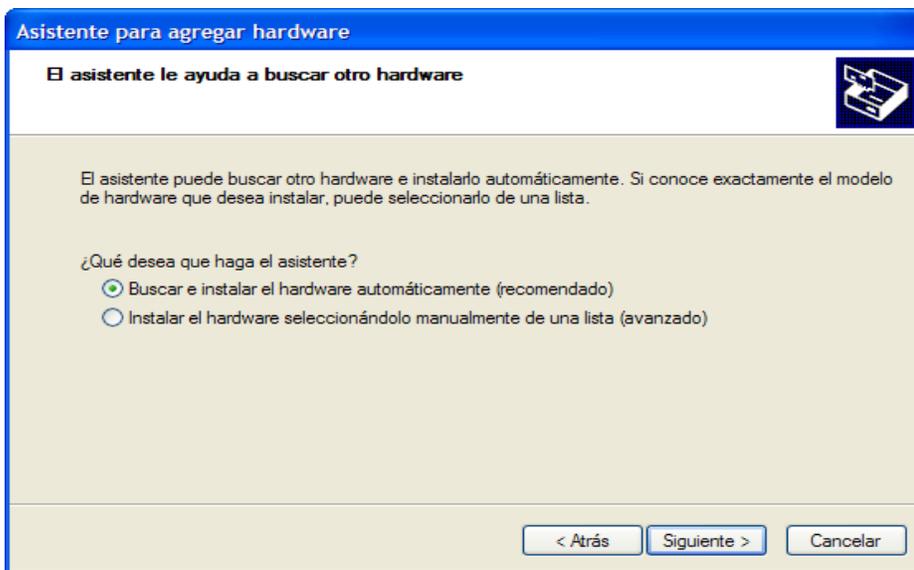
Se selecciona *Sí, ya he conectado el hardware*. Y se aprieta *Siguiente*



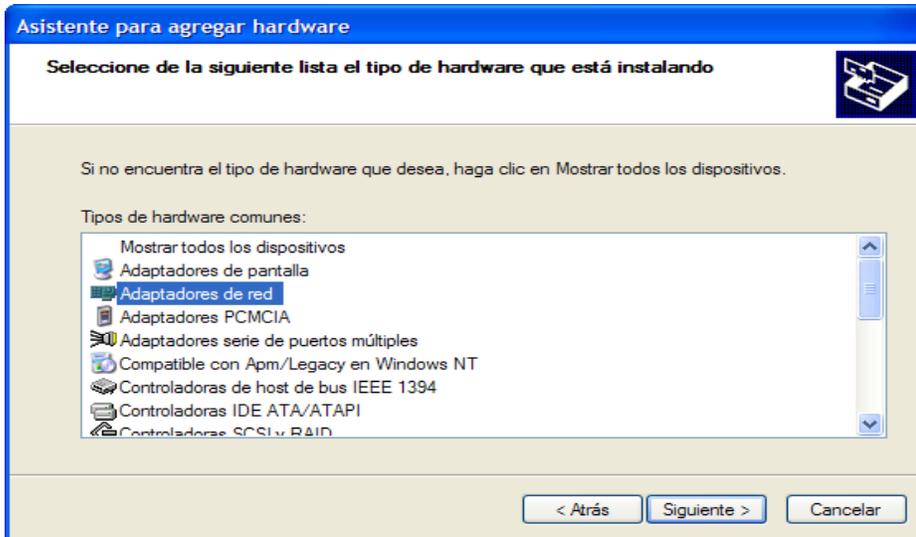
Se desplaza la barra de desplazamiento hasta abajo y se selecciona *agregar un nuevo dispositivo de hardware*.



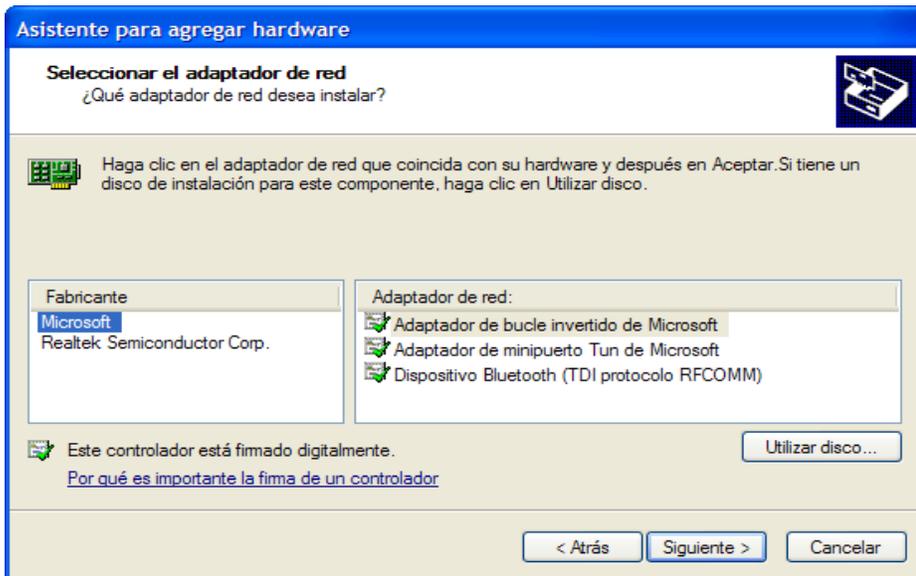
Se aprieta *Siguiente*



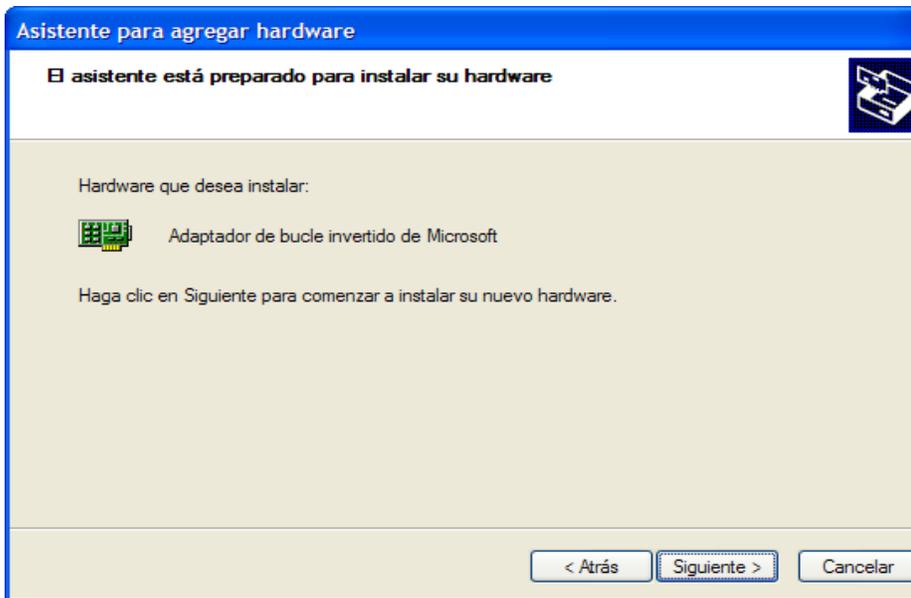
Se selecciona *Instalar el hardware seleccionándolo manualmente de una lista (avanzado)* y se aprieta *Siguiente*.



Se selecciona *Adaptadores de red* y se aprieta *Siguiente*.



Se selecciona fabricante: *Microsoft* y en adaptador de red *Adaptador de bucle invertido*.



Se presiona *Siguiente*.

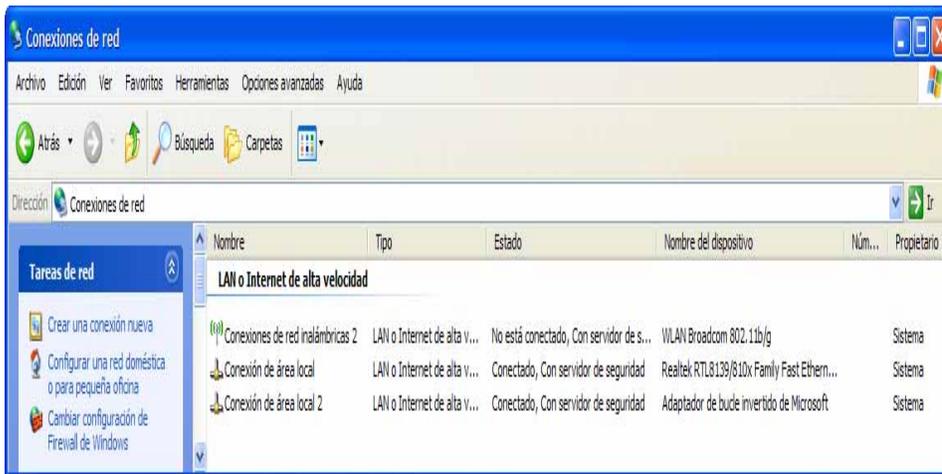


Se presiona *finalizar*.

Para comprobar la instalación del adaptador de bucle invertido.



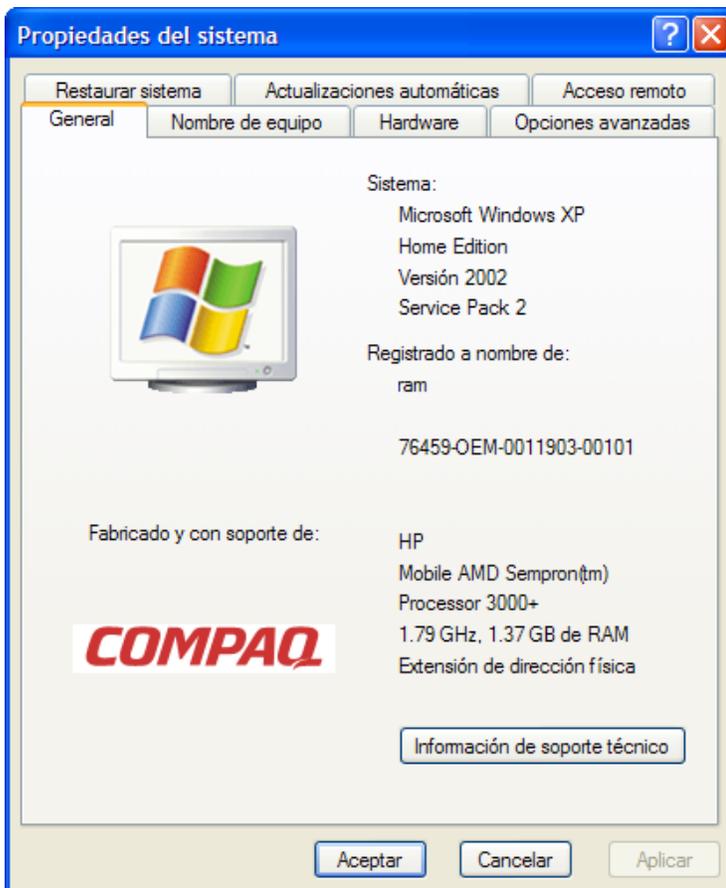
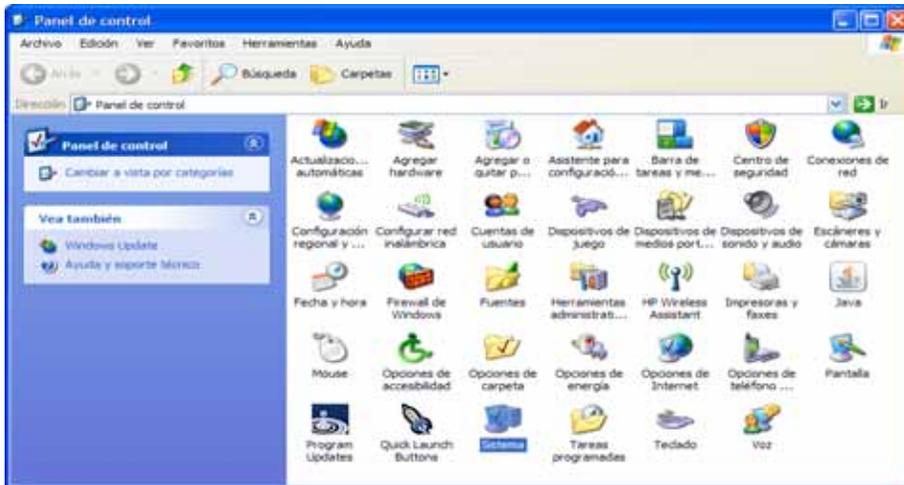
Se aprieta *Conexiones de red* en la ventana de *Panel de control*.



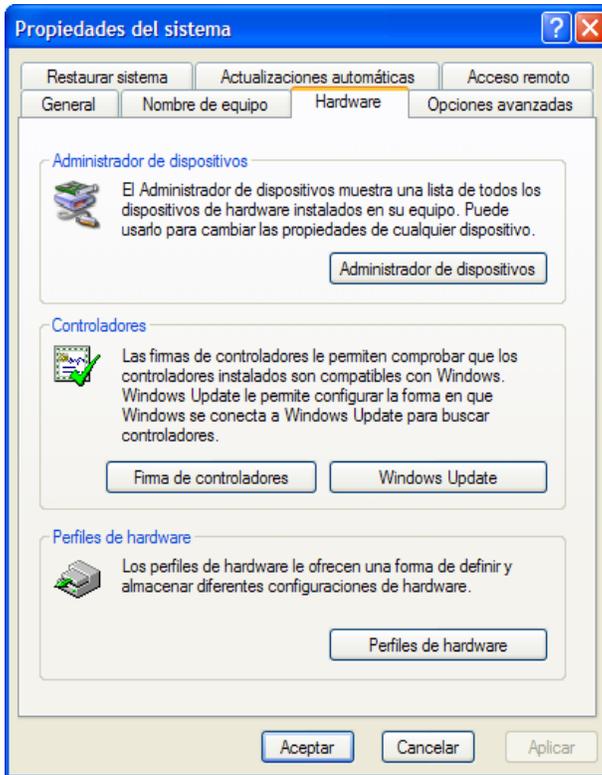
Y se ve el adaptador de bucle invertido.

Si se desea desinstalar; se desinstala como cualquier otro hardware, ya que no podemos desconectarlo físicamente, simplemente se desinstala.

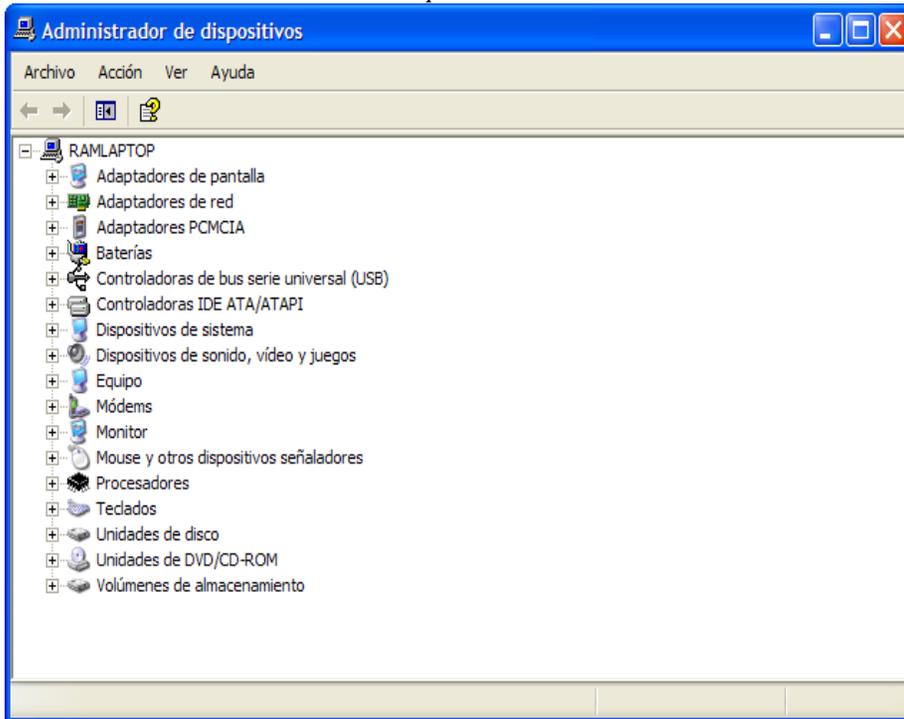
En la ventana de Paneles de control se selecciona Sistema.



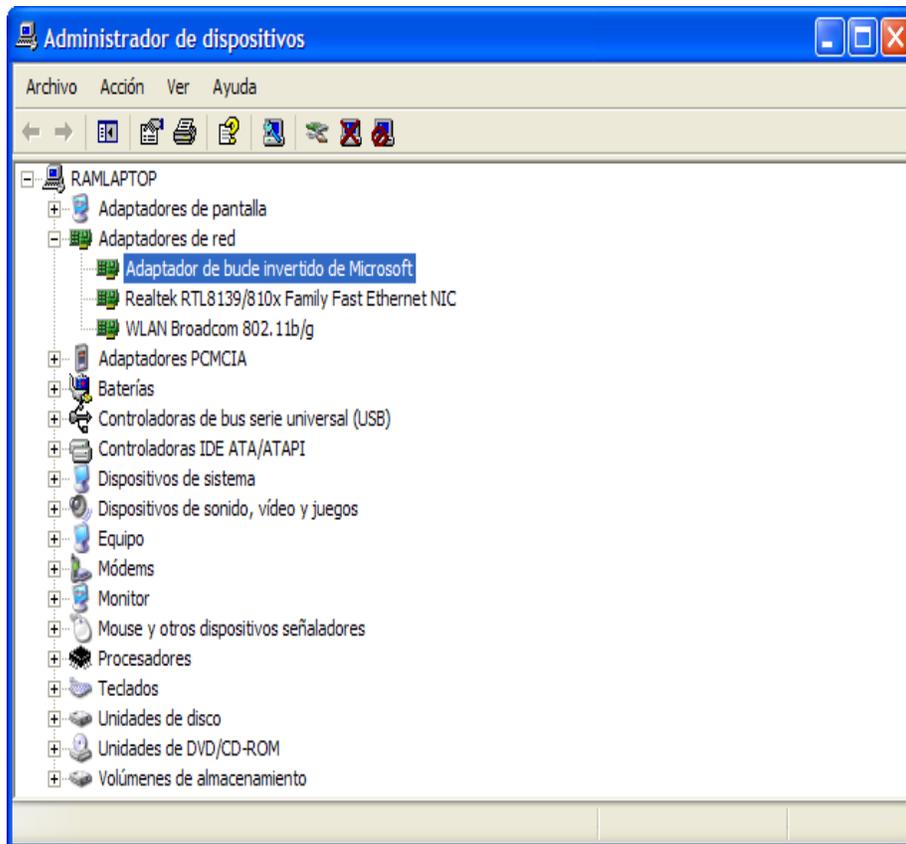
Se selecciona la pestaña de *Hardware*.



Se selecciona *Administrador de dispositivos*.



Se selecciona *Adaptadores de red y Adaptador de bucle invertido*

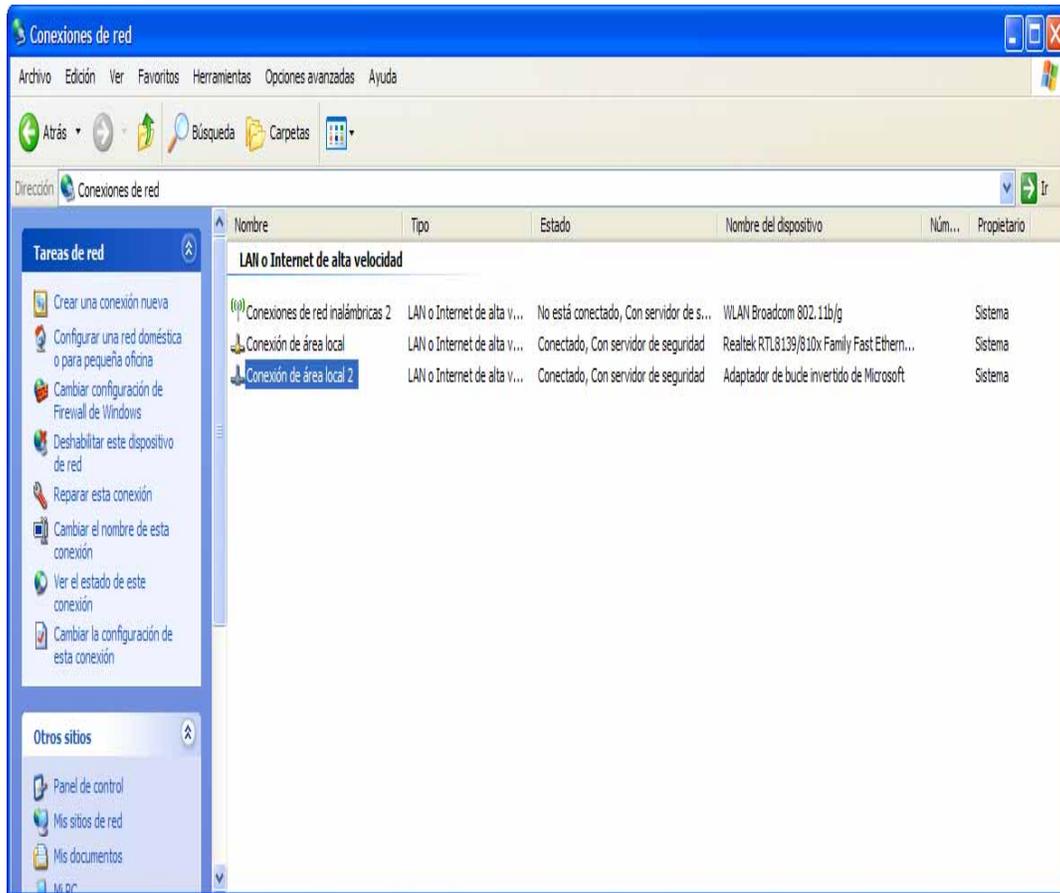


Ahora se presiona en el menú Acción --> Desinstalar. (También puede ser *Botón derecho* sobre el componente y *Desinstalar*)

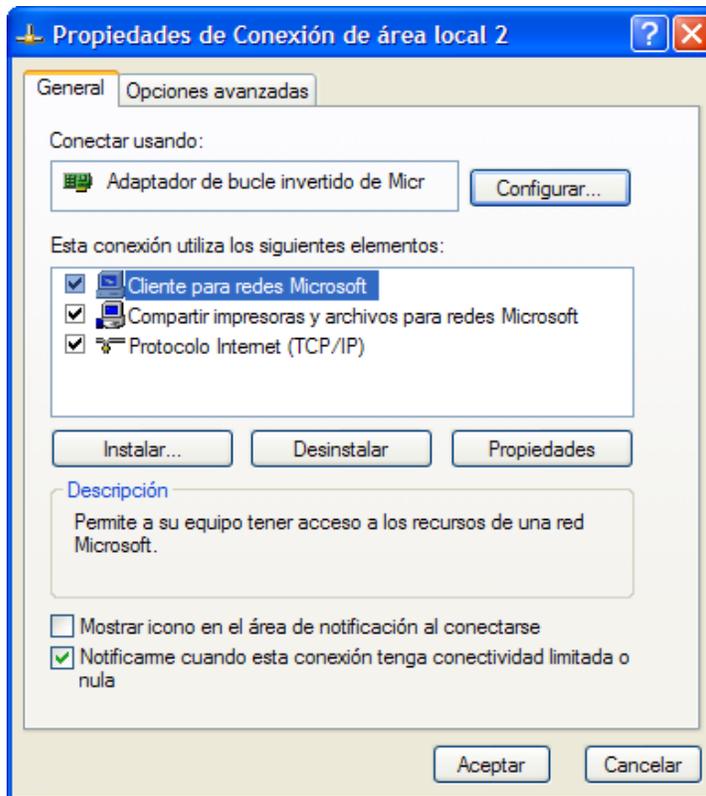
## Parte 2

Finalidad: Configurar Adaptador de bucle invertido de Windows.

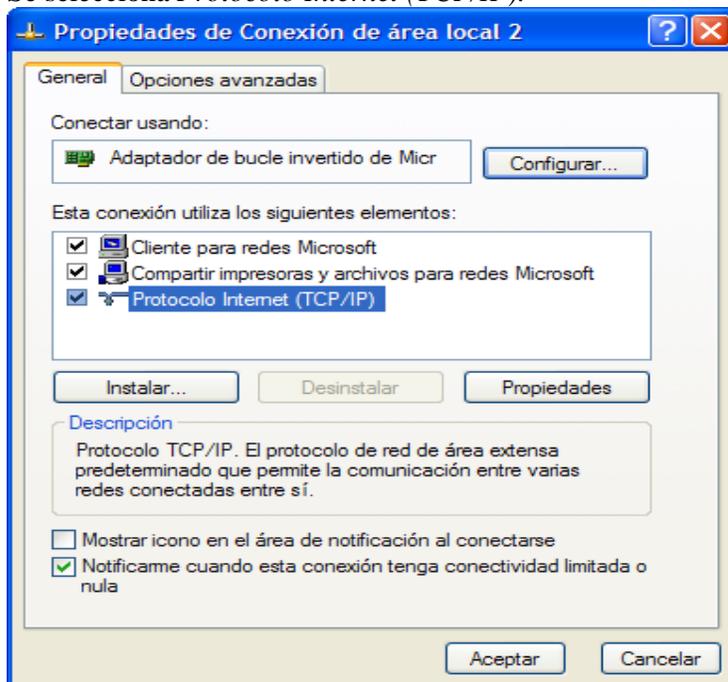
En conexiones de Red.



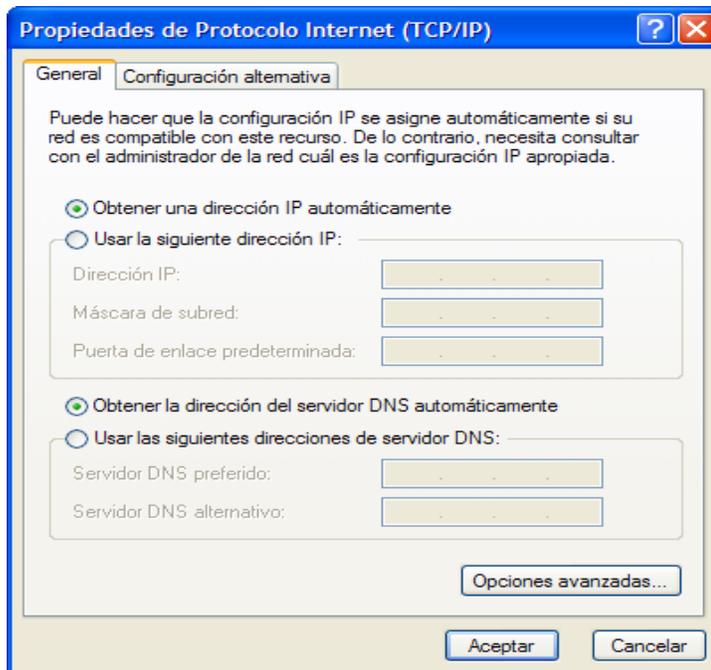
Se selecciona *Adaptadores de red* y *Adaptador de bucle invertido* le invertido, *Botón derecho* del ratón sobre el *Adaptador de bucle invertido* y se selecciona *Propiedades*.



Se selecciona *Protocolo Internet (TCP/IP)*.



Y se selecciona la opción de propiedades.



Se selecciona Usar la siguiente dirección IP

Por ejemplo:

Dirección IP: 10.1.1.2

Máscara de subred: 255.255.255.0

*Aceptar.*



*Cerrar.*

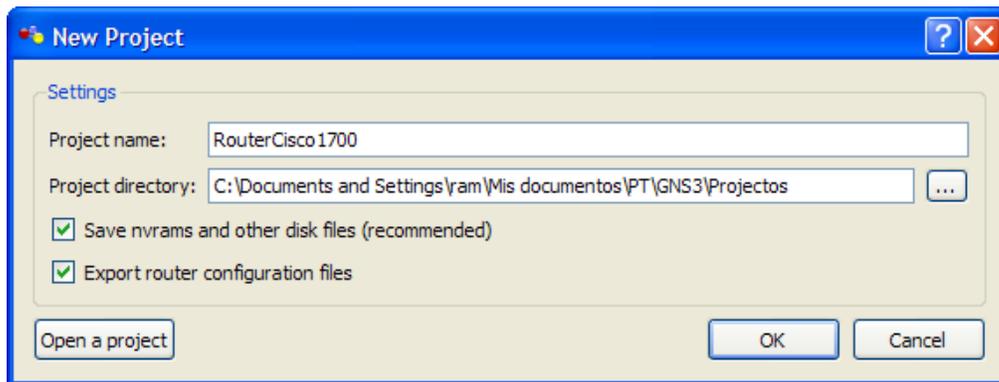
### Parte 3

Finalidad: Arrancar GNS3 y configurar un ejemplo sencillo.

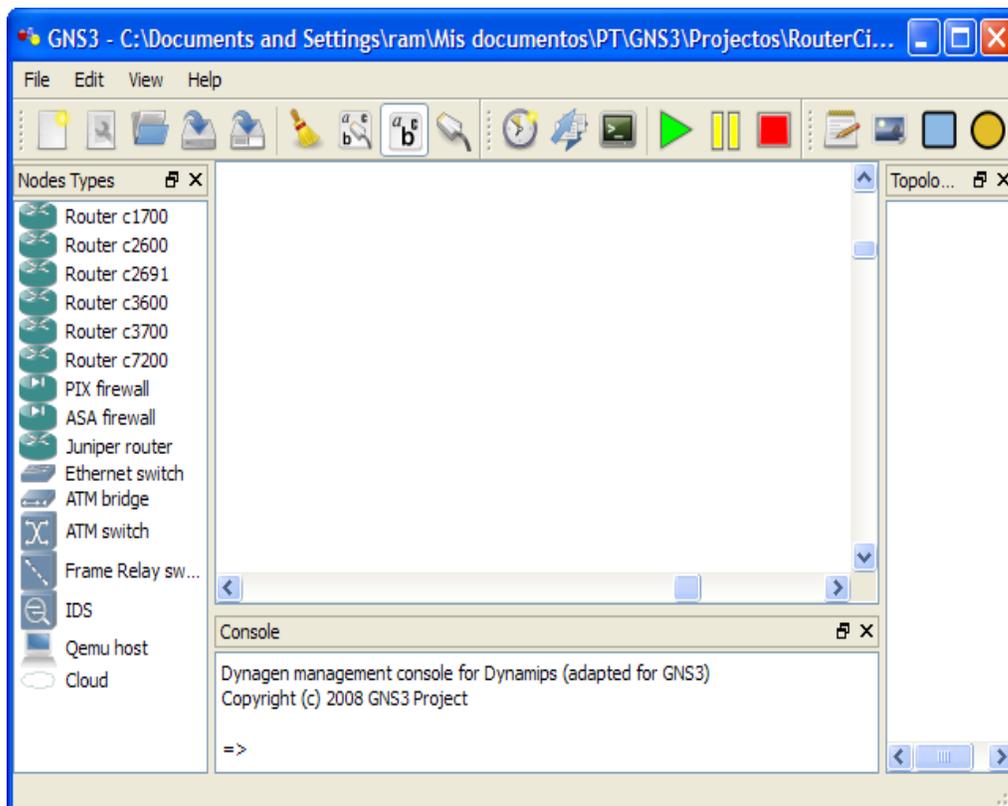
Arrancamos GNS3.

Creamos un proyecto, esto es importante, por que después no tendremos todas las opciones del programa, si no se encuentra dentro e de un proyecto.

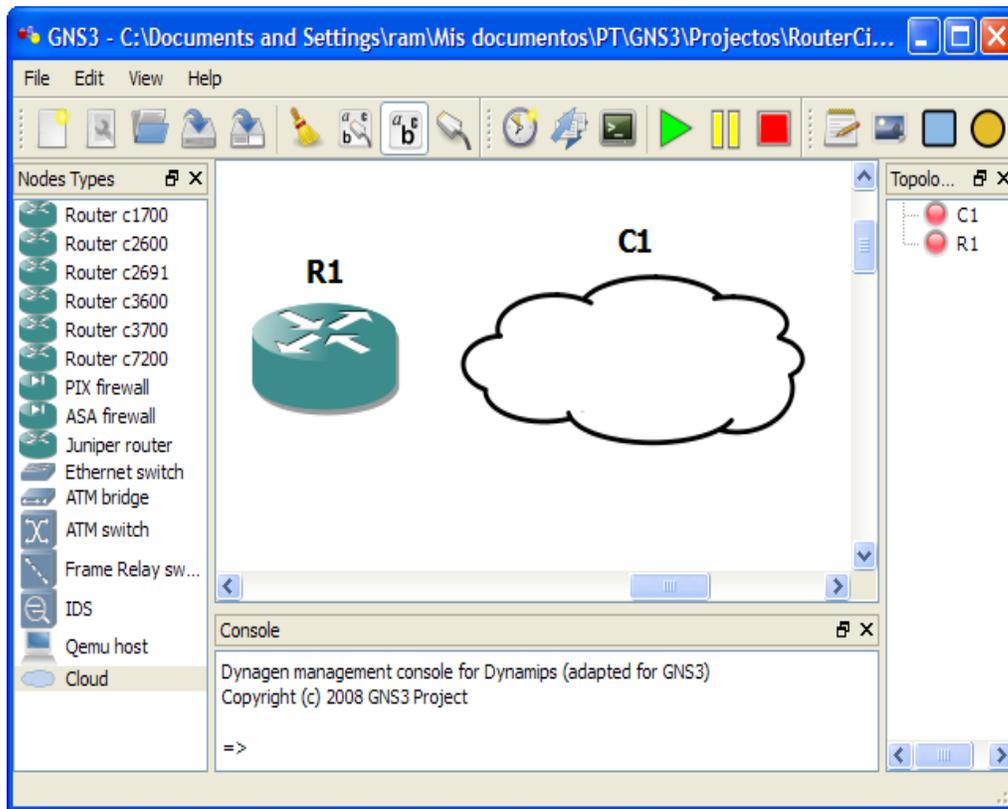
Por ejemplo:



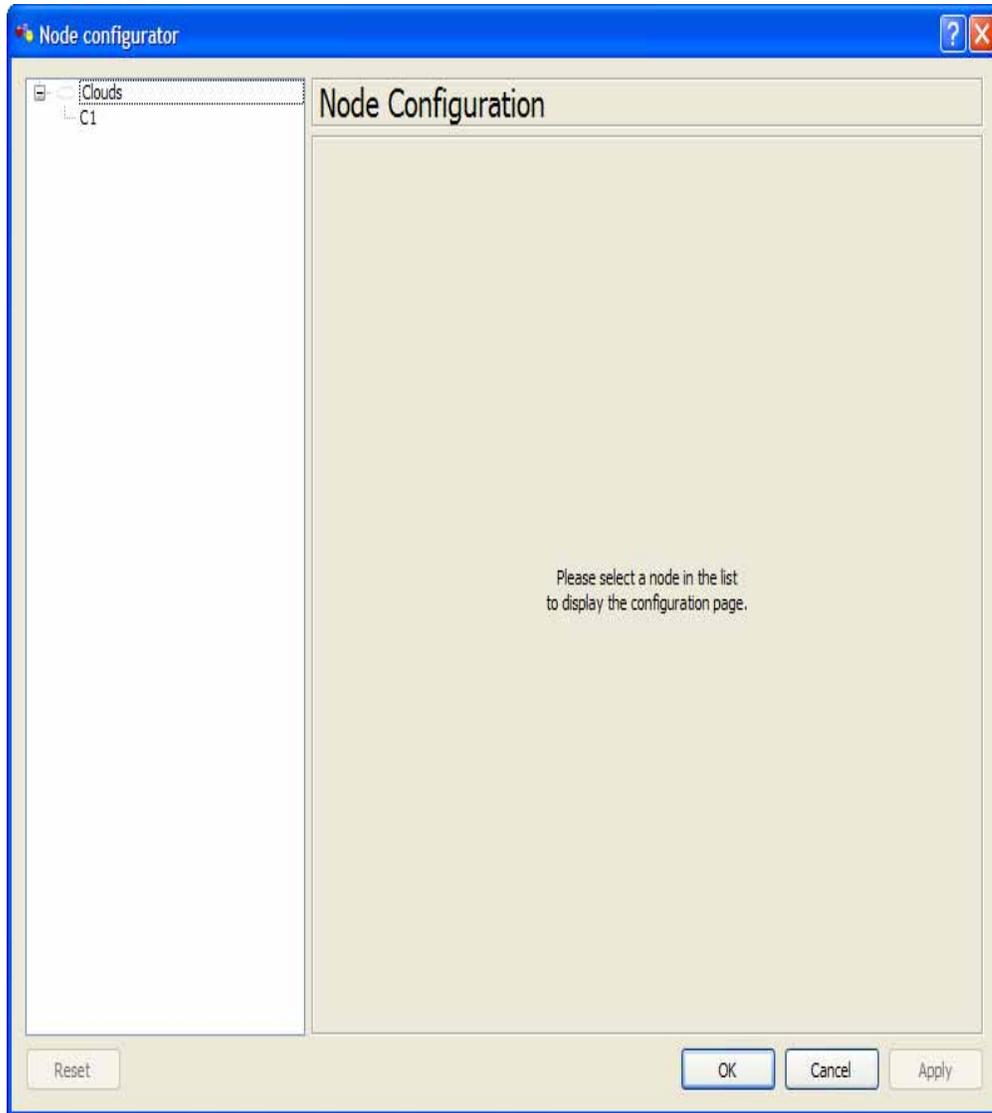
OK



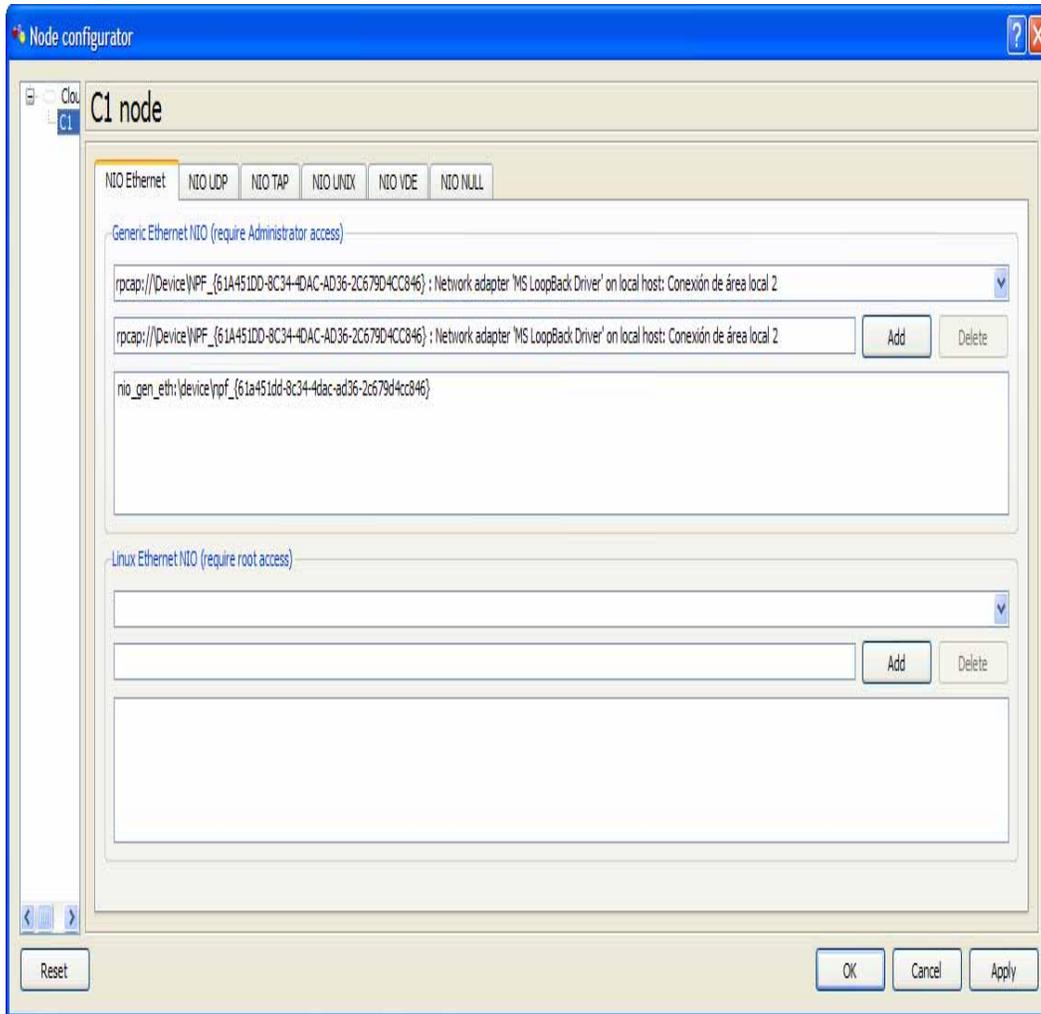
Tomamos y arrastramos un Router (ruteador) Cisco 1700 (previamente configurado) y una Cloud (nube).



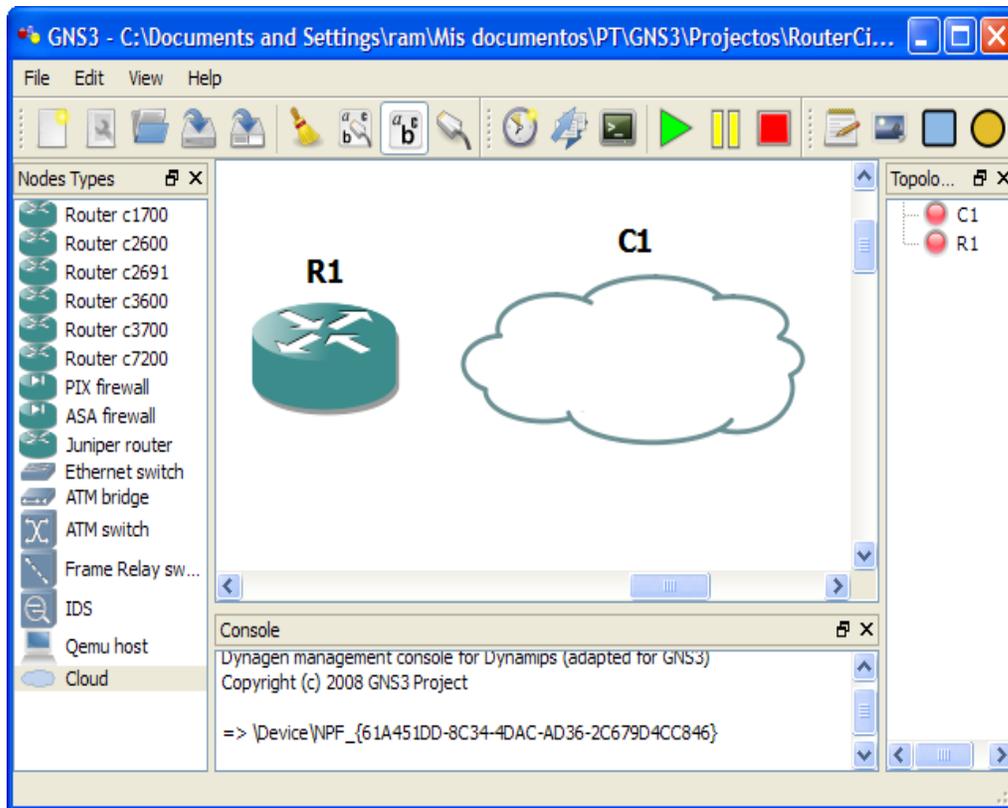
Botón derecho sobre la nube y configurar.



Seleccionamos la nube que acabamos de agregar.



Seleccionamos la pestaña *NIO Ethernet*, donde dice *Generic Ethernet NIO* seleccionamos el *Adaptador de bucle invertido* o *MS Loopback Driver* en inglés y le damos *Add* y luego *OK*



Ahora configuramos el Router (ruteador) añadiéndole una terminal para que tenga donde conectarse el cable.

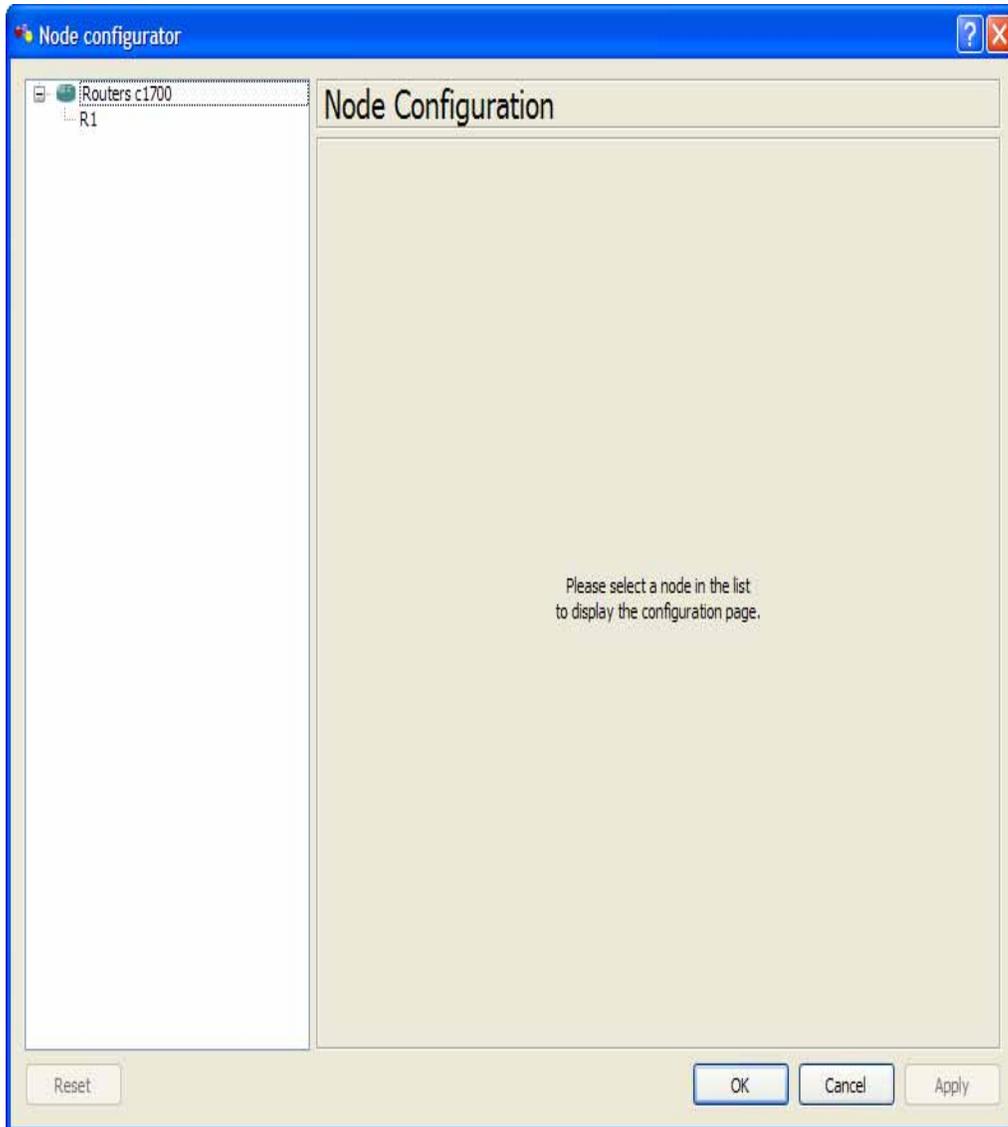
Hay tres opciones:

WIC-1ENET es igual a una interfase e.

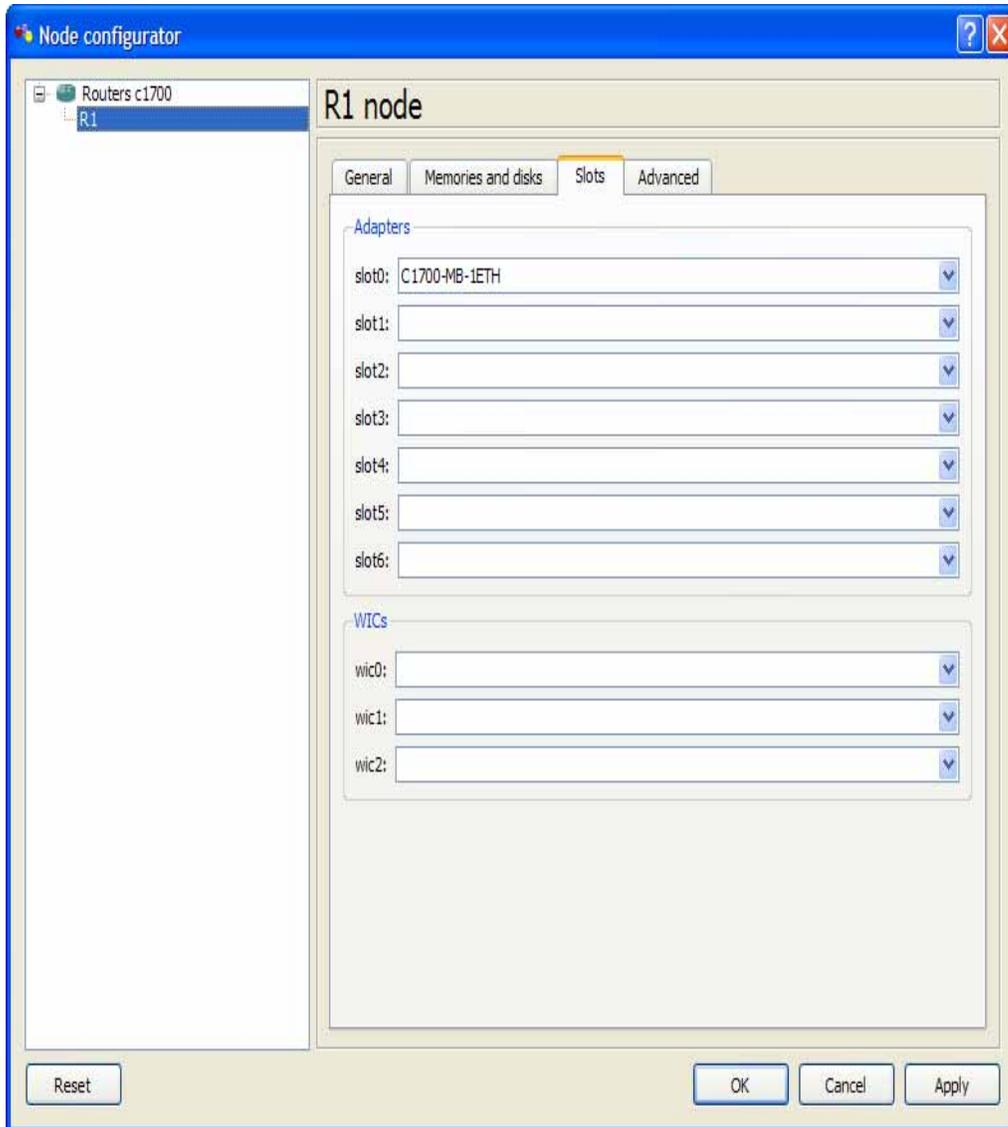
WIC-1T es igual a una interfase s.

WIC-2T es igual a dos interfases s.

*Botón derecho* sobre el router (ruteador) y *Configurar*.



Seleccionamos *R1* y luego la pestaña *Slots*.

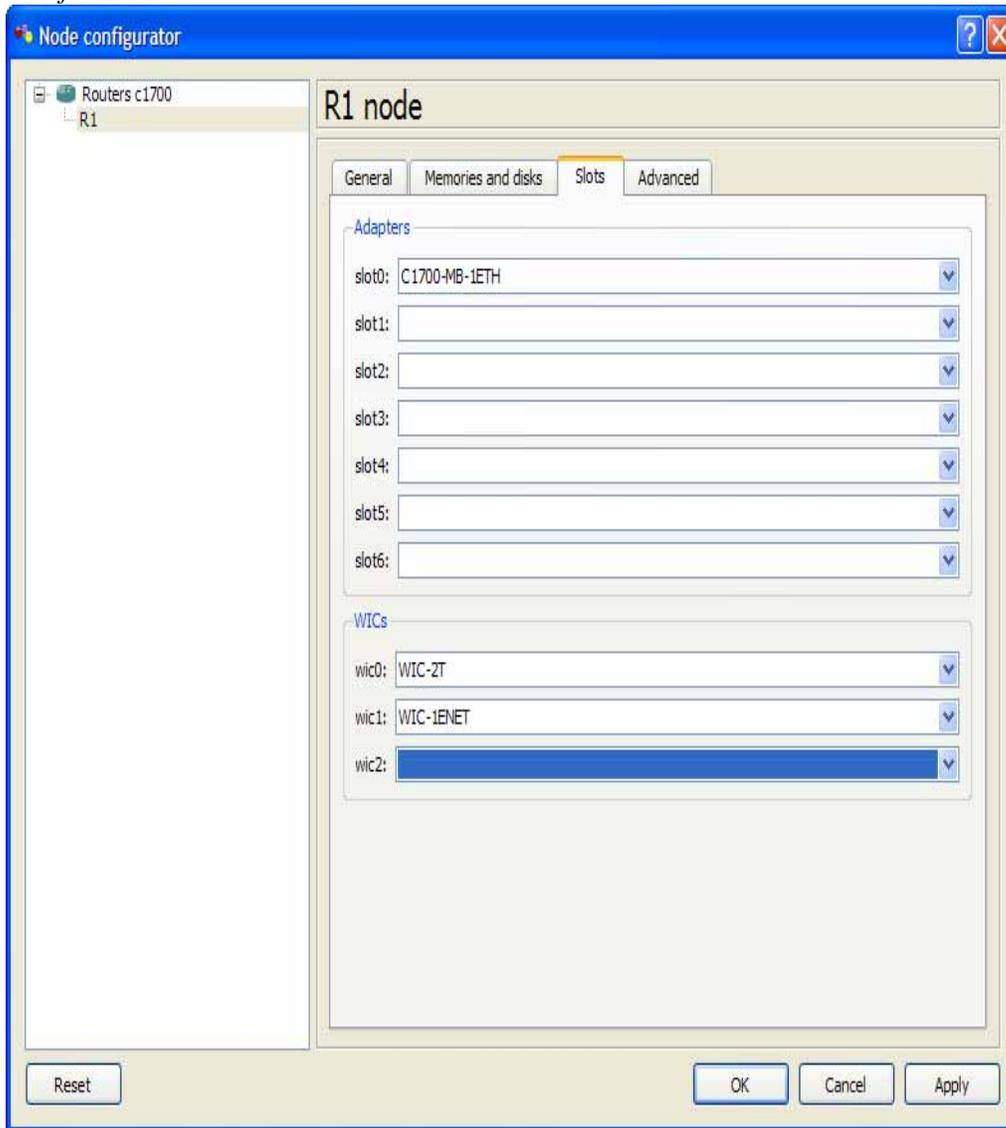


En WICs

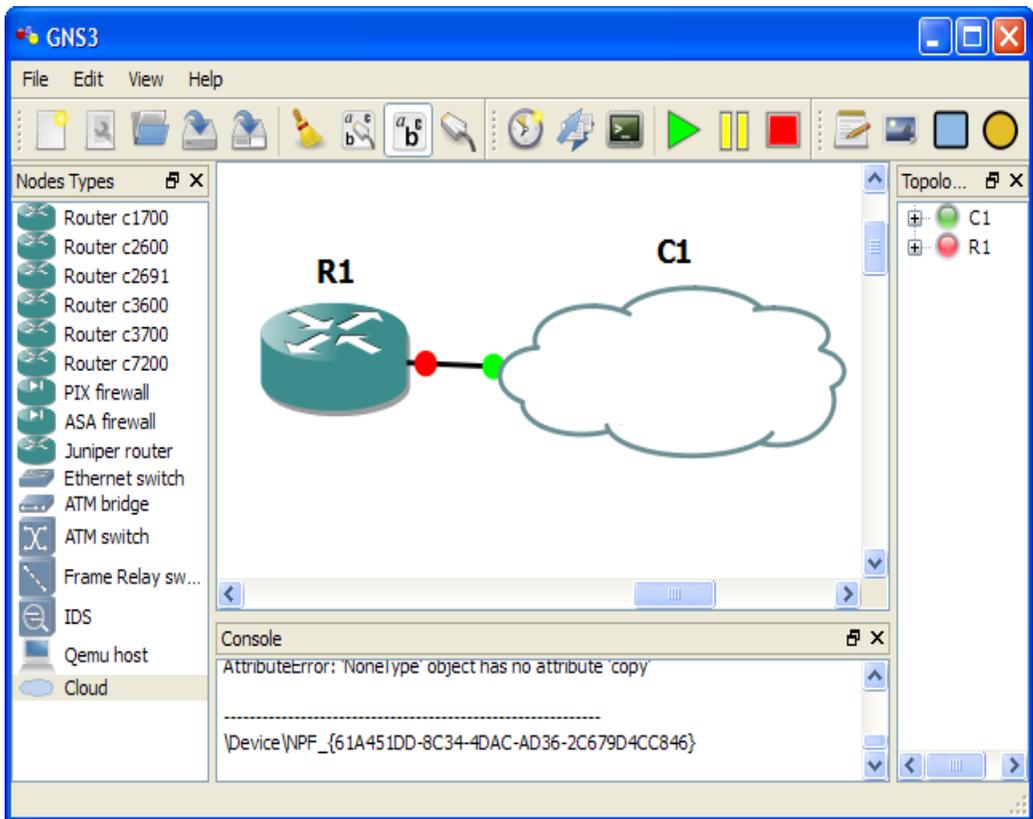
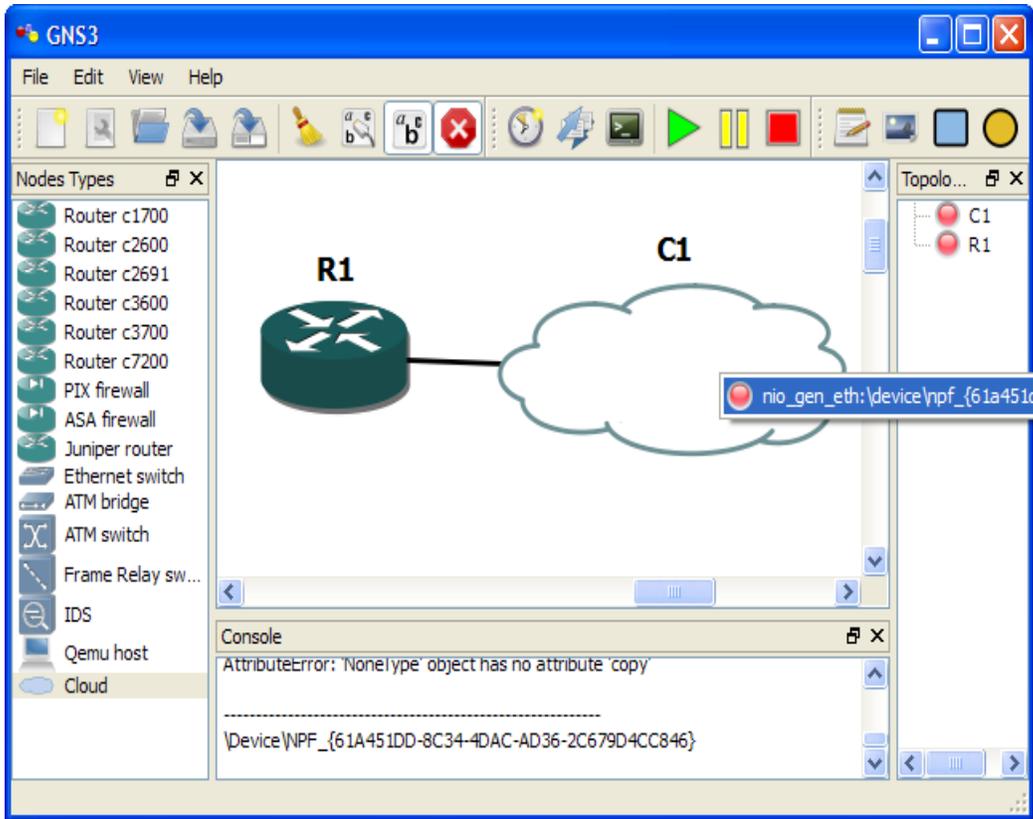
Seleccionamos WIC-1ENETy WIC-2T.

*OK*

De regreso en la interfase.



De regreso en la interfase principal seleccionamos la herramienta *Add link* (el que parece un extremo de un RJ45 sin letras) seleccionamos *FastEthernet* y lo extendemos desde el router (ruteador) hasta la nube, hay que tomar en cuenta que en la nube hay que escoger que Adaptador vamos a usar, pero como solo tenemos uno la elección no es mayor problema.



Empezamos la emulación, dejamos que cargue el sistema operativo del router (ruteador) y abrimos una consola al router (ruteador).

Ahora los siguientes comandos:

```
Router (ruteador)>enable
```

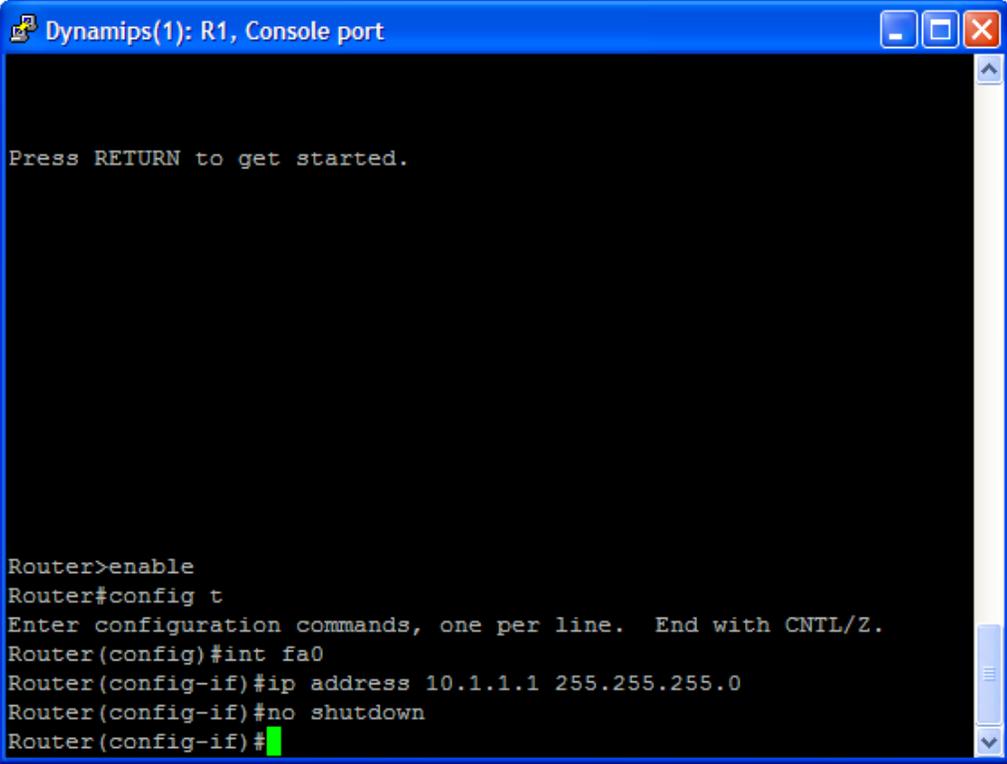
```
Router (ruteador)#config t
```

```
Router (ruteador)(config)# ( y esperamos unos instantes si es la primera vez sale una línea extra después de unos instantes )
```

```
Router (ruteador)(config)# int fa0
```

```
Router (ruteador)(config-if)#ip address 10.1.1.1 255.255.255.0
```

```
Router (ruteador)(config-if)#no shutdown
```



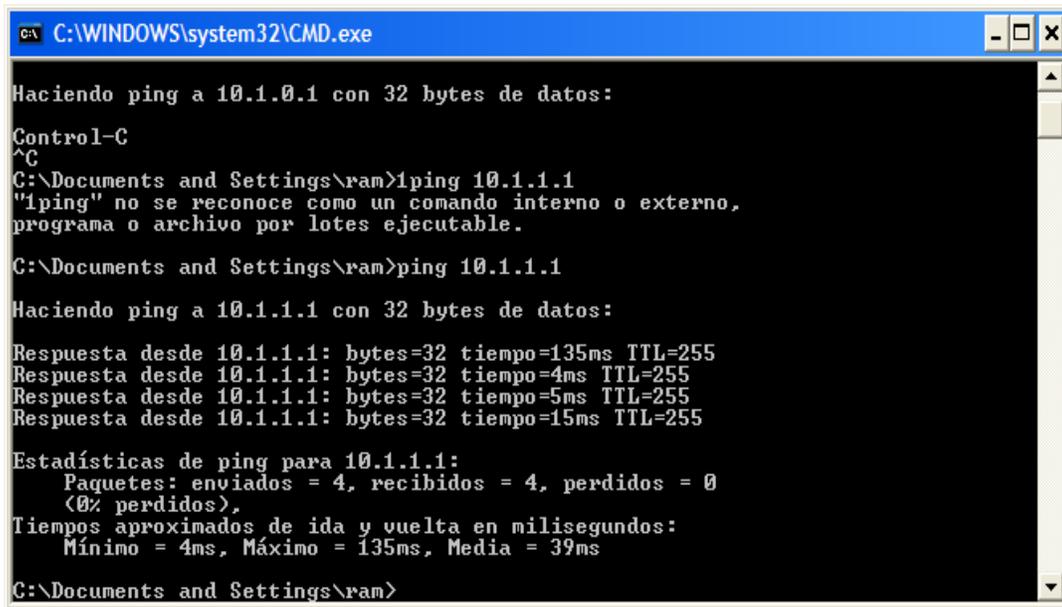
```
Dynamips(1): R1, Console port

Press RETURN to get started.

Router>enable
Router#config t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#int fa0
Router(config-if)#ip address 10.1.1.1 255.255.255.0
Router(config-if)#no shutdown
Router(config-if)#
```

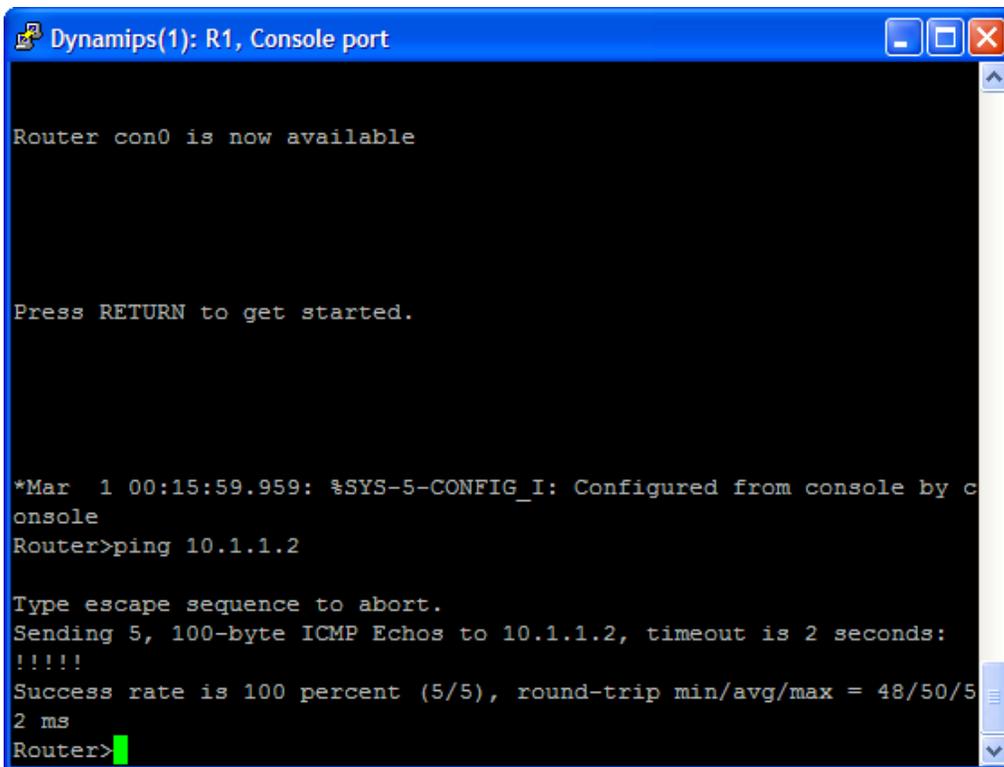
Abrimos una terminal de Windows.

Y lanzamos un *ping 10.1.1.1*



```
C:\WINDOWS\system32\CMD.exe
Haciendo ping a 10.1.0.1 con 32 bytes de datos:
Control-C
^C
C:\Documents and Settings\ram>ping 10.1.1.1
"ping" no se reconoce como un comando interno o externo,
programa o archivo por lotes ejecutable.
C:\Documents and Settings\ram>ping 10.1.1.1
Haciendo ping a 10.1.1.1 con 32 bytes de datos:
Respuesta desde 10.1.1.1: bytes=32 tiempo=135ms TTL=255
Respuesta desde 10.1.1.1: bytes=32 tiempo=4ms TTL=255
Respuesta desde 10.1.1.1: bytes=32 tiempo=5ms TTL=255
Respuesta desde 10.1.1.1: bytes=32 tiempo=15ms TTL=255
Estadísticas de ping para 10.1.1.1:
    Paquetes: enviados = 4, recibidos = 4, perdidos = 0
              (0% perdidos),
    Tiempos aproximados de ida y vuelta en milisegundos:
        Mínimo = 4ms, Máximo = 135ms, Media = 39ms
C:\Documents and Settings\ram>
```

También podemos lanzar un *ping 10.1.1.2* desde la consola solo para estar seguros.



```
Dynamips(1): R1, Console port
Router con0 is now available
Press RETURN to get started.
*Mar 1 00:15:59.959: %SYS-5-CONFIG_I: Configured from console by console
Router>ping 10.1.1.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.1.2, timeout is 2 seconds:
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 48/50/52 ms
Router>
```

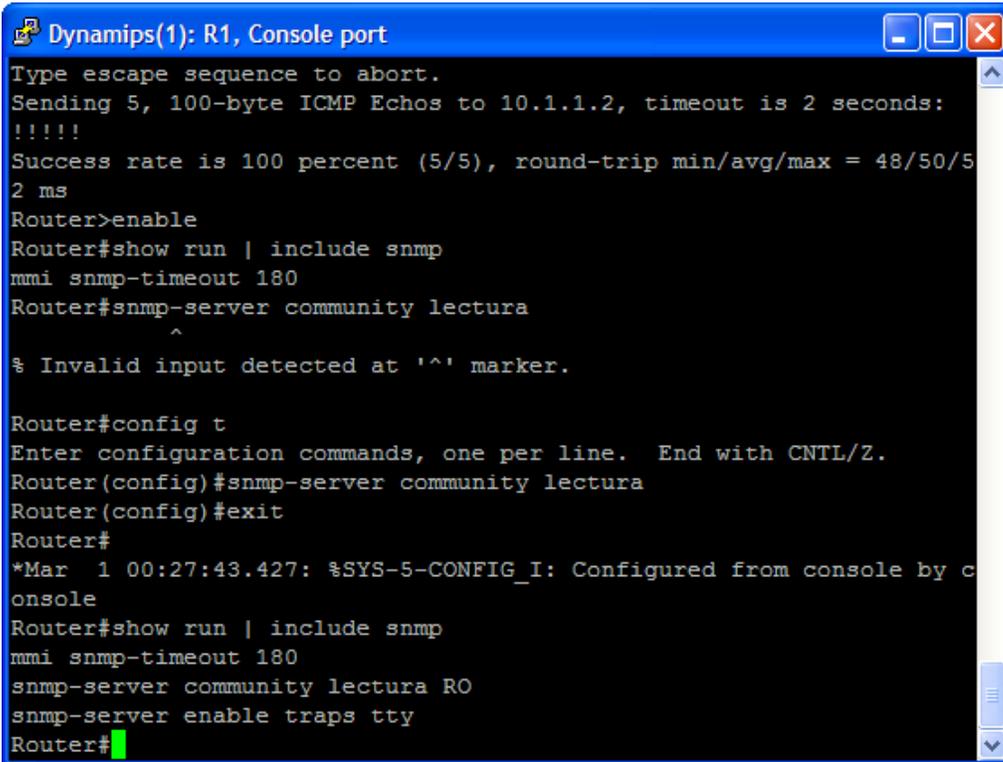
Nota: una manera de hacer que GNS3 consuma menos recursos, aparte del uso de IDLEPC, es descomprimir la imagen del IO del router (ruteador), renombrarla con el mismo nombre que el .bin y cambiar la extensión a .image.

#### Parte 4

Finalidad: Habilitar SNMP en el router (ruteador).

En términos prácticos tecleamos el siguiente comando:

Router (ruteador)(config)# snmp-server community lectura



```
Dynamips(1): R1, Console port
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.1.2, timeout is 2 seconds:
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 48/50/52 ms
Router>enable
Router#show run | include snmp
mmmi snmp-timeout 180
Router#snmp-server community lectura
^
% Invalid input detected at '^' marker.

Router#config t
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#snmp-server community lectura
Router(config)#exit
Router#
*Mar 1 00:27:43.427: %SYS-5-CONFIG_I: Configured from console by console
Router#show run | include snmp
mmmi snmp-timeout 180
snmp-server community lectura RO
snmp-server enable traps tty
Router#
```

## Parte 5

Finalidad: Configurar router (ruteador) para tener acceso desde un TELNET por ethernet y no por serial.

Nota: esto disminuye el uso de recursos por parte de GNS3.

Conviene habilitar el uso de contraseña.

```
Router (ruteador) > enable
```

```
Router (ruteador) # configure terminal
```

```
Router (ruteador) (config) #
```

```
Router (ruteador) (config) # enable password <contraseña>
```

Y también configuramos para tener acceso desde una sesión de TELNET por ethernet.

```
Router (ruteador) > enable
```

```
Password?:
```

```
Router (ruteador) # configure terminal
```

```
Router (ruteador) (config) # line vty 0 1 ( se pueden tener dos sesiones simultáneas de TELNET )
```

```
Router (ruteador) (config-line) # password <contraseña>
```

```
Router (ruteador) (config-line) # exit
```

Y salvamos la configuración interna en el router (ruteador)

copy running-config startup-config y después un simple enter.

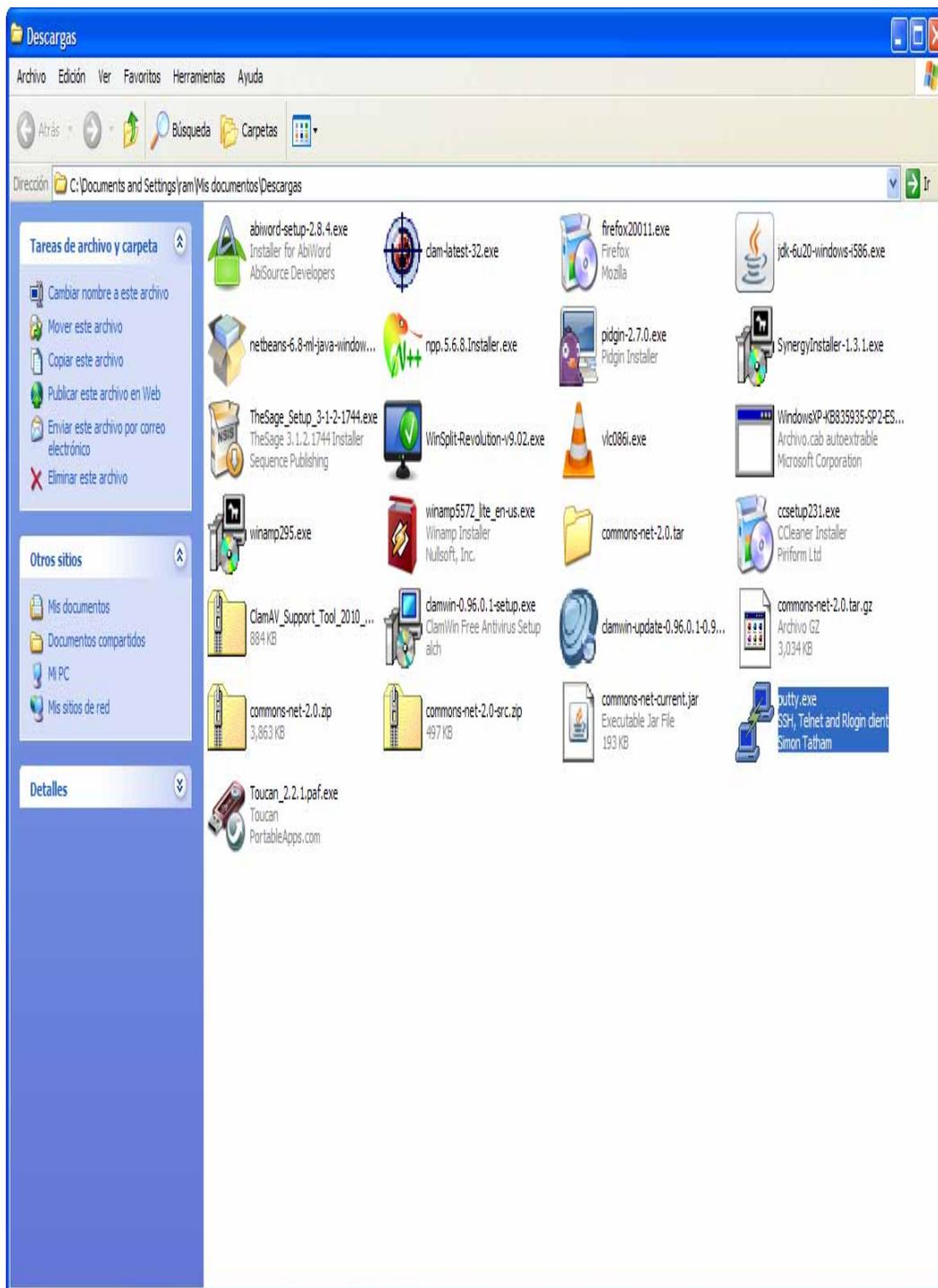
```
Dynamips(0): R1, Console port

Router>enable
Router#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#config t
      ^
% Invalid input detected at '^' marker.

Router(config)#enable password secreto
Router(config)#line vty 0 1
Router(config-line)#password otrosecreto
Router(config-line)#exit
Router(config)#exit
Router#
*Mar  1 00:02:52.563: %SYS-5-CONFIG_I: Configured from console by console
Router#copy running-config startup-config
Destination filename [startup-config]? y
%Error copying nvram:y (Invalid argument)
Router#copy running-config startup-config
Destination filename [startup-config]? yes
%Error copying nvram:yes (Invalid argument)
Router#copy running-config startup-config
Destination filename [startup-config]?
Building configuration...
[OK]
Router#
```

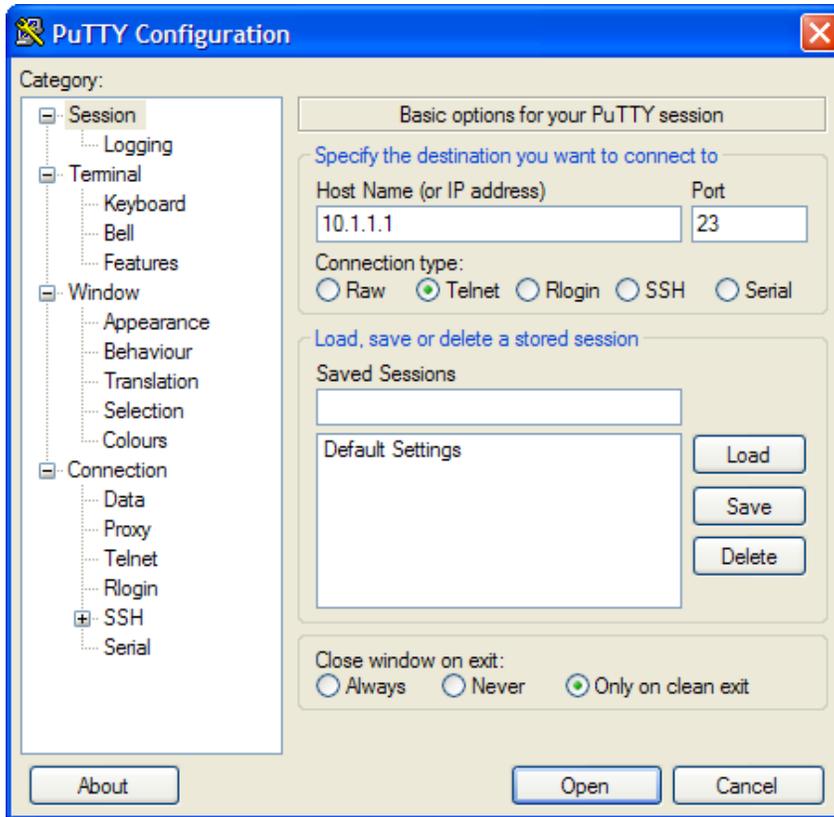
Cerramos la ventana.

Descargamos putty

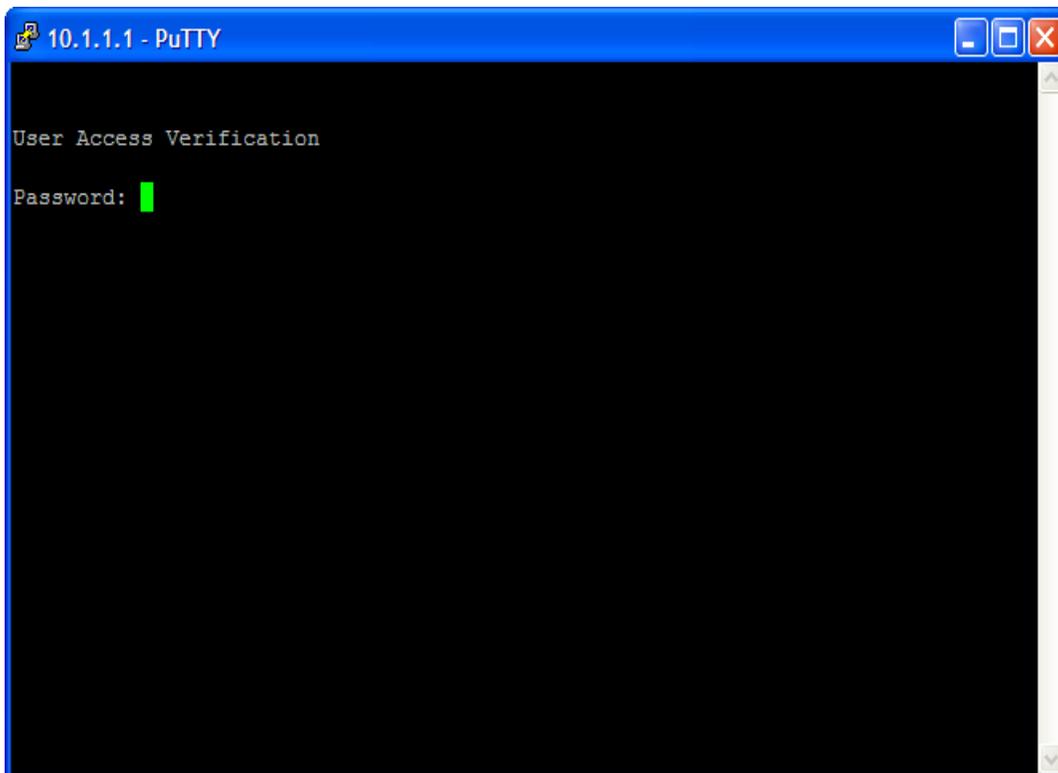


y simplemente damos doble click, no instala simplemente corre.

Damos la dirección *10.1.1.1* y el protocolo *TELNET*.



Seleccionamos *Open*.



Proporcionamos la contraseña.



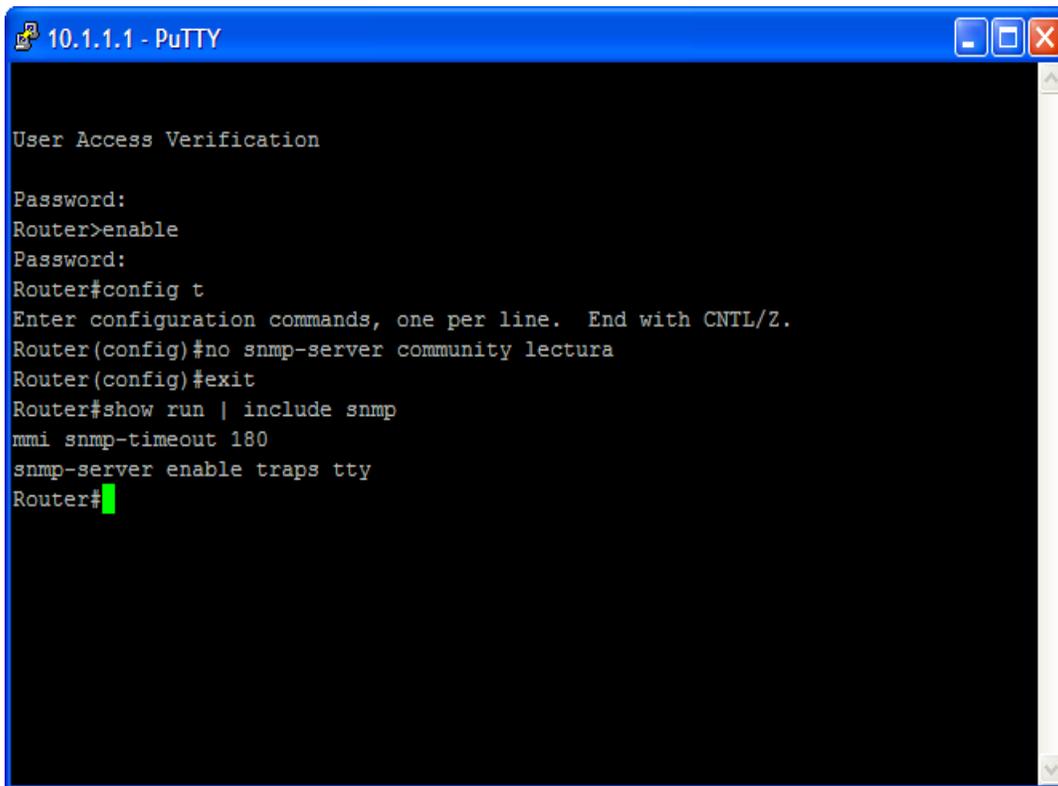
y listo tenemos acceso a través del ethernet y no del serial. Necesario si necesitamos mandar algún comando a través de algún lenguaje de programación.

## Parte 6

Finalidad: Configurar snmp más a fondo para tener acceso a snmp desde algún lenguaje de programación.

Primero borramos la comunidad de lectura que existía previamente.

Router (ruteador)(config)# no snmp-server community lectura



```
10.1.1.1 - PuTTY
User Access Verification
Password:
Router>enable
Password:
Router#config t
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#no snmp-server community lectura
Router(config)#exit
Router#show run | include snmp
mmi snmp-timeout 180
snmp-server enable traps tty
Router#
```

Hay que generar una vista para establecer a que MIBS va a tener acceso.

Router (ruteador)(config)# snmp-server view vista <alcance> included

El alcance puede ser por ejemplo: cisco, system o una OID( 1.3.6.\* o iso.org.dod.\* ).

Router (ruteador)(config)# snmp-server view vista 1.\* included



```
10.1.1.1 - PuTTY
User Access Verification
Password:
Password:
Router>enable
Password:
Router#config t
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#snmp-server view vista 1.*
% Incomplete command.

Router(config)#snmp-server view vista 1.* included
Router(config)#
```

Ahora creamos una lista de acceso

```
Router (ruteador)(config)# access-list <#> permit host <ip>
```

En nuestro caso:

```
Router (ruteador)(config)# access-list 1 permit host 10.1.1.2
```

```
Router (ruteador)(config)# access-list 1 deny any log
```

Permite la IP 10.1.1.2 y niega el acceso a todas las demás IP's.



The image shows a PuTTY terminal window titled "10.1.1.1 - PuTTY". The terminal output is as follows:

```
User Access Verification
Password:
Router>enable
Password:
Router#config t
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#access-list 1 permit host 10.1.1.2
Router(config)#access-list 1 deny any log
Router(config)#
```

Ahora creamos las dos comunidades, la de lectura y la de escritura por simplicidad.

```
Router (ruteador)(config)# snmp-server community <nombre> view <vista> <tipo>
```

```
Router (ruteador)(config)# snmp-server community lectura view vista ro
```



```
10.1.1.1 - PuTTY
User Access Verification

Password:
Router>enable
Password:
Router#config t
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#snmp-server community lectura view vista ro
Router(config)#
```

Después la comunidad de escritura.

```
Router (ruteador)(config)# snmp-server community <nombre> view <vista> <tipo>
<lista>
```

```
Router (ruteador)(config)# snmp-server community escritura rw 1
```

```
10.1.1.1 - PuTTY

User Access Verification

Password:
Router>enable
Password:
Router#config t
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#snmp-server community lectura view vista ro
Router(config)#
Router(config)#snmp-server community escritura rw 1
Router(config)#
```

Para confirmar

```
Router (ruteador)# show access-lists
```

```
Router (ruteador)# show run | include snmp
```

```
10.1.1.1 - PuTTY

User Access Verification

Password:
Router>enable
Password:
Router#config t
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#snmp-server community lectura view vista ro
Router(config)#
Router(config)#snmp-server community escritura rw 1
Router(config)#exit
Router#show access-lists
Standard IP access list 1
 10 permit 10.1.1.2
 20 deny any log
Router#show run | include snmp
mmi snmp-timeout 180
snmp-server view vista iso.* included
snmp-server community lectura view vista RO
snmp-server community escritura RW 1
snmp-server enable traps tty
Router#
```

## Casos de uso

### Caso de uso 001: Autenticación.

#### Actores

- Usuario
- Aplicación

#### Descripción

- Esta interfaz de usuario se sirve para gestionar la conexión al dispositivo indicado, la autenticación ya sea sólo con contraseña o con usuario y contraseña y la autenticación como administrador del equipo.

#### Pre-condiciones

- Ninguna.

#### Post-condiciones

- La aplicación ha establecido una conexión con el Router (ruteador).
- El usuario se ha autenticado a través del nombre de usuario y su contraseña, o la simple conexión se ha autenticado con sólo la contraseña.
- El usuario se autenticado como administrador del Router (ruteador).

#### Flujo normal.

1. La aplicación muestra la interfaz de autenticación con todos sus elementos deshabilitados.
2. La aplicación habilita las opciones para conexión.
3. La aplicación muestra los protocolos disponibles (telnet) para conexión.
4. La aplicación muestra un campo de captura para la dirección. *IP* (protocolo de Internet) o el nombre del dispositivo del Router (ruteador).
5. La aplicación muestra el botón de "conectar".
6. El usuario selecciona el protocolo con el cual se desea conectar.
7. El usuario proporciona la *IP* (protocolo de internet) o el nombre del Host (anfitrión) al cual se desea conectar.
8. El usuario aprieta el botón de "conectar".
9. La aplicación habilita las opciones para la autenticación de la conexión.
10. La aplicación deshabilita las opciones de conexión.
11. La aplicación habilita el campo de captura de nombre usuario.
12. La aplicación habilita el campo de captura de contraseña.
13. La aplicación habilita el botón de "autenticar".
14. El usuario proporciona el nombre de usuario.
15. El usuario proporciona la contraseña.
16. El usuario presiona el botón de "autenticar".
17. La aplicación deshabilita las opciones de autenticación.
18. La aplicación habilita el campo de captura de contraseña como administrador.
19. La aplicación habilita el botón de "habilitar".
20. El usuario proporciona la contraseña de administrador nivel 15.
21. El usuario aprieta el botón de "habilitar".
22. La aplicación deshabilita todos los elementos.
23. La aplicación será la interfaz de autenticación.

24. Termina caso de uso.

#### Flujos alternativos.

8. El usuario no proporcionó un Host (anfitrión).
  - La aplicación marca un mensaje de error.
  - La aplicación regresa al paso 6.
8. El usuario proporciona un Host (anfitrión) inválido.
  - La aplicación marca un mensaje de error.
  - La aplicación regresa al paso 6.
10. La aplicación no habilita el campo de captura de nombre de usuario.
  - La aplicación habilita el campo de captura de contraseña.
  - La aplicación habilita el botón de "autenticar".
  - El usuario proporciona la contraseña.
  - El usuario presiona el botón de "autenticar".
  - La aplicación va a paso 17.
16. El usuario no proporcionó un nombre de Host (anfitrión) y/o una contraseña.
  - La aplicación marca un mensaje de error.
  - La aplicación va a paso 11
16. El usuario proporciona un nombre de Host (anfitrión) y/o una contraseña inválidos.
  - La aplicación marca un mensaje de error.
  - La aplicación va a paso 11
21. El usuario no proporcionó una contraseña de nivel administrador.
  - La aplicación marca un mensaje de error.
  - La aplicación va a paso 17
21. El usuario proporcionó una contraseña incorrecta de nivel administrador.
  - La aplicación marca un mensaje de error.
  - La aplicación va a paso 17

#### Vista.

The image shows a graphical user interface for configuring a network connection. It is divided into three numbered sections:

- 1 Escoge tu protocolo (Telnet) y escribe el nombre del Host:** This section contains two radio buttons: "Telnet" (which is selected) and "SSH". To the right, there is a text input field labeled "Host" containing the IP address "10.1.1.1". Below these elements is a "Conectar" button.
- 2 Proporciona lo siguiente:** This section contains two text input fields: "Usuario" with the value "hector" and "Contraseña" with five dots. Below these is an "Autenticar" button.
- 3 Contraseña de administrador ( enable ):** This section contains a single text input field labeled "Contraseña" with five dots. Below it is a "Habilitar" button.

**Notas y asunto:**

- El protocolo SSH no se encuentra implementando ya que su implementación está fuera de los alcances de este proyecto.

## Caso de uso 002: Información general.

### Actores

- Usuario
- Aplicación

### Descripción

- Muestra la información general del equipo en un campo de texto.

### Pre-condiciones

- El usuario deberá estar autenticado como administrador del sistema.

### Post-condiciones

- Ninguna.

### Flujo normal.

1. La aplicación muestra el nombre de Host (anfitrión) del dispositivo.
2. La aplicación muestra un campo de texto en blanco.
3. La aplicación muestra el botón de abrir con mi mostrar información".
4. El usuario presiona el botón de "mostrar información".
5. La aplicación muestra una ventana emergente con la leyenda "Recuperando información del Router (ruteador), puede tardar unos instantes." y el botón de aceptar.
6. El usuario presiona el botón de aceptar.
7. La aplicación carga el campo de texto con toda la información del Router (ruteador).
8. Termina caso de uso.

### Flujos alternativos.

- Ninguno

## Vista.

The screenshot displays a software window with a blue title bar and standard Windows window controls. The interface is divided into several sections:

- Navigation Tabs:** Privilegios, Mensajes, Auto asegurar, SNMP, SSH, NTP, Vistas, ACL, Auto asegurar 2. Below these are sub-tabs: Información General (selected), Contraseña, Líneas, Usuarios, Conexion, Acceso, Niveles.
- Dispositivo: Router:** A list of memory addresses and their associated components:

0x80782C30	0000013056	0000000001	0000013056	DHCPD Message Workspace
0x80782C4C	0000008192	0000000001	0000008192	DHCPD Workspaces
0x8078563C	000000276	0000000001	000000276	DHCPD Radix Information Nodes
0x8078563C	0000001500	0000000001	0000001500	DHCPD Radix Information Nodes
0x80785758	0000000128	0000000001	0000000128	DHCPD Workspaces
0x80794900	0000001028	0000000001	0000001028	DHCPD Database Workspace
0x807B3724	0000065536	0000000001	0000065536	ddb counters struct
0x807CCC94	0000002492	0000000001	0000002492	dialer_ckt_sw_t_pool
0x807D172C	0000002000	0000000001	0000002000	DIALER FWD Requests
0x807D8814	0000000092	0000000001	0000000092	Init
0x807D8838	0000000132	0000000001	0000000132	Init
0x807D9BC0	0000000040	0000000001	0000000040	CLID Group
0x807D9BE4	0000000024	0000000001	0000000024	CG:Name
0x807D9C1C	0000000032	0000000001	0000000032	CLID Entry
0x807D9C54	0000000024	0000000001	0000000024	CE:Num
0x807F8F04	0000000144	0000000008	0000000144	Entity MIB API
0x807F8F04	0000000188	0000000001	0000000188	Entity MIB API
0x807F8F04	0000000192	0000000001	0000000192	Entity MIB API
0x807F8F04	0000000204	0000000001	0000000204	Entity MIB API
0x807F8F04	0000000024	0000000001	0000000024	(fragment) (Free Blocks)
0x807F8F04	0000000048	0000000001	0000000048	(fragment) (Free Blocks)
0x807F8F04	0000000092	0000000001	0000000092	(fragment) (Free Blocks)
0x807FF068	0000000024	0000000003	0000000024	Init
0x807FF090	0000000040	0000000003	0000000040	Init
0x80800238	0000000256	0000000002	0000000256	Init
0x8080893C	0000000960	0000000004	0000000960	Fair Queueing
0x8080894C	0000001536	0000000002	0000001536	Fair Queueing
- Mostrar información:** A button located below the list.
- Tareas a realizar:** A list of tasks: reconectar, guardar, desconectar.
- Ejecutar:** A button with an empty text input field next to it.
- Desconectar:** A button.

## Notas y asunto:

- Ninguno.

## Caso de uso 02: Contraseña.

### Actores

- Usuario
- Aplicación

### Descripción

- Esta interfaz muestra si la contraseña "password" se encuentra habilitada o deshabilitada asimismo da la opción de cambiar o des habilitar la misma.

### Pre-condiciones

- El usuario se encuentra autenticado dentro de la aplicación.
- El usuario ha seleccionado la pestaña de contraseña.

### Post-condiciones

- La contraseña sea habilitado o deshabilitado.

### Flujo normal habilitar.

1. La aplicación muestra una etiqueta con el estado actual de la contraseña como deshabilitada.
2. La aplicación muestra las opciones de: habilitar y deshabilitar.
3. La aplicación habilita la opción de “habilitar”.
4. La aplicación deshabilita la opción de “deshabilitar”.
5. La aplicación muestra un campo de mensajes en blanco.
6. La aplicación muestra un cuadro de captura de texto.
7. La aplicación muestra el botón de añadir.
8. El usuario selecciona la opción de habilitar.
9. El usuario proporciona la nueva contraseña.
10. El usuario presiona el botón de añadir.
11. La aplicación añade la tarea a la lista de tareas por hacer.
12. Termina caso de uso.

### Flujos alternativos habilitar.

9. El usuario no proporciona una nueva contraseña.

- La aplicación marca un mensaje de error.
- La aplicación regresa al paso 8.

### Flujo normal deshabilitar.

1. La aplicación muestra una etiqueta con el estado actual de la contraseña como habilitada.
2. La aplicación muestra las opciones de: habilitar y deshabilitar.
3. La aplicación habilita ambas opciones
4. La aplicación muestra un campo de mensajes en blanco.
5. La aplicación muestra un cuadro de captura de texto.
6. La aplicación muestra el botón de añadir.
7. El usuario selecciona la opción de deshabilitar.
8. El usuario presiona el botón de añadir.
9. La aplicación añade la tarea a la lista de tareas por hacer.
10. Termina caso de uso.

### Flujos alternativos deshabilitar.

7. No existe otra manera de que se autentifica que el administrador ante el Router (ruteador).

- La aplicación llena el área de mensajes con la leyenda de que no existe otra manera de autentificar el administrador.
- Va a paso 10.

Nota: Se exhorta que antes de deshabilitar la autenticación por contraseña (password) se habilite la autenticación a través de contraseña secreto (secret) ya que de lo contrario se perderá la conexión con el Router (ruteador).

7. El usuario selecciona la opción de habilitar.

- La aplicación llena el área de mensajes solicitando la nueva contraseña.
- El usuario proporciona la nueva contraseña.
- Va a paso ocho.

Vista.

Privilegios Mensajes Auto asegurar SNMP SSH NTP Vistas ACL Auto asegurar 2  
Información General **Contraseña** Lineas Usuarios Conexion Acceso Niveles

Password **Habilitado**

Si deseas establecer un password nuevo:

Habilitar

Deshabilitar

-

**Añadir**

Tareas a realizar

reconectar  
guardar  
desconectar

**Ejecutar**  **Desconectar**

Notas y asunto:  
Ninguno.

## Caso de uso 003: Gestión de líneas.

### Actores

- Usuario
- Aplicación

### Descripción

- Esta interfaz muestra las líneas de conexión así como las contraseñas necesarias para su acceso, también permite establecer o modificar las contraseñas existentes.

### Pre-condiciones

- El usuario está autenticado dentro del sistema.
- El usuario seleccionó la pestaña de líneas.

### Post-condiciones

- Una línea ha establecido o modificado su contraseña.

### Flujo normal.

1. La aplicación muestra una tabla con todas las líneas existentes así como su contraseña.
2. La aplicación muestra un cuadro de selección con todas las líneas existentes.
3. La aplicación muestra un campo de captura de texto para la nueva contraseña.
4. La aplicación muestra el botón con la opción de añadir.
5. El usuario selecciona la línea que quiera modificar.
6. El usuario proporciona la contraseña.
7. El usuario selecciona el botón de añadir.
8. La aplicación añade la tarea a la lista de tareas ejecutar.
9. Termina caso de uso.

### Flujos alternativos.

7. El usuario no proporciona una nueva contraseña.
  - La aplicación marca un mensaje de error.
  - La aplicación regresa al paso 6.

Vista.

Linea	Contraseña
con 0	none
aux 0	none
vtv 0	linea
vtv 1	linea
vtv 2	jijiji
vtv 3	hola
vtv 4	adios

Tareas a realizar

- reconectar
- guardar
- desconectar

Ejecutar      Desconectar

**Notas y asunto:**

No se provee la opción de eliminar una contraseña ya que iría contra las buenas prácticas de configuración de un Router (ruteador).

## Caso de uso 004: Gestión de usuarios.

### Actores

- Usuario
- Aplicación

### Descripción

- Esta interfaz permite agregar nuevos usuarios y eliminarlos.

### Pre-condiciones

- El usuario está autenticado dentro de la aplicación.
- El usuario seleccionó la pestaña de usuarios.

### Post-condiciones

- se ha dado de alta o de baja a un usuario dentro de la configuración del Router (ruteador).

### Flujo normal.

1. La aplicación muestra una tabla de usuarios, con los usuarios existentes dentro del Router (ruteador).
2. La aplicación muestra un botón con la opción de eliminar.
3. La aplicación muestra un campo de captura del nombre de usuario.
4. La aplicación muestra un campo de selección con los tipos de encriptación disponibles para la contraseña.
5. La aplicación muestra un campo de captura de texto para la contraseña de ese usuario.
6. La aplicación muestra una barra de desplazamiento para escoger el nivel de usuario.
7. El usuario proporciona un nuevo nombre de usuario.
8. El usuario selecciona el tipo de encriptación para la contraseña.
9. El usuario provee la contraseña para su nuevo usuario.
10. El usuario provee el nivel de privilegios para ese usuario.
11. El usuario aprieta el botón de añadir.
12. La aplicación añade esa tarea a la lista de tareas a ejecutar
13. Termina caso de uso.

### Flujos alternativos.

7. El usuario selecciona uno de los renglones de la tabla de usuarios.
  - El usuario presiona el botón de eliminar.
  - La aplicación añade la tarea de eliminar el usuario seleccionado a la lista de tareas a ejecutar.
  - Termina caso de uso.
11. El usuario no proporciona un nombre de usuario.
  - La aplicación marca un mensaje de error.
  - La aplicación regresa al paso 7.
11. El usuario no proporciona una contraseña.
  - La aplicación marca un mensaje de error.
  - La aplicación regresa al paso 7.

Vista.

The screenshot shows a web-based configuration interface for a network device. The top navigation bar includes tabs for Privilegios, Mensajes, Auto asegurar, SNMP, SSH, NTP, Vistas, ACL, and Auto asegurar 2. Below this, a sub-menu contains Información General, Contraseña, Lineas, Usuarios (selected), Conexion, Acceso, and Niveles.

The main content area features a table with the following data:

Usuario	Nivel	Encriptación
hector	15	secret
ramiro	15	password
pepe	6	secret
mm	12	password

Below the table is an "Eliminar" button. The form below contains input fields for "Usuario", "Encriptación" (a dropdown menu currently set to "password"), and "Contraseña". A "Nivel de privilegios" section shows a slider set to "1". An "Añadir" button is located at the bottom left of the main form area.

At the bottom, a "Tareas a realizar" section lists "reconectar", "guardar", and "desconectar". Below this are "Ejecutar" and "Desconectar" buttons, with an empty input field between them.

Notas y asunto:  
Ninguno.

## Caso de uso 005: Gestión de vistas.

### Actores

- Usuario
- Aplicación

### Descripción

- Esta interfaz gestiona el tiempo que una interfaz mantiene viva una sesión del Router (ruteador).

### Pre-condiciones

- El usuario esta autenticado dentro del sistema.
- El usuario selecciona la opción de conexión.

### Post-condiciones

- El tiempo (minutos y segundos) de conexión máxima de una línea cambia en la configuración del Router (ruteador).

### Flujo normal.

1. La aplicación muestra un campo de selección con las líneas existentes en el Router (ruteador).
2. La aplicación muestra un selector de número para los minutos.
3. La aplicación muestra un selector de número para segundos.
4. La aplicación muestra un botón de añadir.
5. La aplicación muestra una tabla con la información de las líneas existentes.
6. El usuario selecciona la línea.
7. El usuario selecciona los minutos.
8. El usuario selecciona los segundos.
9. El usuario selecciona la opción de añadir.
10. La aplicación agrega esta tarea a la lista de tareas a ejecutar.
11. Termina caso de uso.

### Flujos alternativos.

- Ninguno.

Vista.

The screenshot shows a network configuration window with a blue title bar and standard window controls. The interface has a tabbed menu at the top with the following tabs: Privilegios, Mensajes, Auto asegurar, SNMP, SSH, NTP, Vistas, ACL, Auto asegurar 2, Información General, Contraseña, Líneas, Usuarios, Conexión, Acceso, and Niveles. The 'Conexión' tab is active. Below the tabs, the 'Línea' section features a dropdown menu set to 'con 0'. Underneath are two spinners for 'Minutos' and 'Segundos', both set to '0'. An 'Añadir' button is positioned above a table. The table has three columns: 'Línea', 'Minutos', and 'segundos'. It lists configurations for 'con 0', 'aux 0', and 'vty' lines 0 through 4. Below the table is a large empty rectangular area. At the bottom, the 'Tareas a realizar' section contains a list of actions: 'reconectar', 'guardar', and 'desconectar'. At the very bottom, there are 'Ejecutar' and 'Desconectar' buttons, with an empty text input field between them.

Línea	Minutos	segundos
con 0	2	2
aux 0	4	4
vty 0	1	1
vty 1	default min	default sec
vty 2	3	2
vty 3	default min	default sec
vty 4	default min	default sec

Notas y asunto:  
Ninguno.

## Caso de uso 006: Intentos por dispositivo.

### Actores

- Usuario
- Aplicación

### Descripción

- Esta interfaz se permite cambiar el número máximo de intentos por dispositivo Router (ruteador) desde otro dispositivo, antes de que el Router (ruteador) bloquee temporalmente el dispositivo.

### Pre-condiciones

- El usuario esta autenticado dentro de sistema.
- El usuario selecciona la pestaña de acceso.

### Post-condiciones

- El número de intentos por dispositivo Router (ruteador) se ha modificado dentro de la configuración del Router (ruteador).

### Flujo normal.

1. La aplicación muestra el número de intentos establecidos dentro del sistema.
2. La aplicación muestra un selector para el número de intentos.
3. La aplicación muestra un botón de añadir.
4. El usuario selecciona el nuevo número de intentos.
5. El usuario presiona el botón de añadir.
6. La aplicación añade la tarea a la lista de tareas a ejecutar.
7. Termina caso de uso.

### Flujos alternativos.

- Ninguno

Vista.

Privilegios Mensajes Auto asegurar SNMP SSH NTP Vistas ACL Auto asegurar 2  
Información General Contraseña Lineas Usuarios Conexion Acceso Niveles

Número de intentos establecidos: 2

2

Añadir

Tareas a realizar

reconectar  
guardar  
desconectar

Ejecutar Desconectar

Notas y asunto:  
Ninguna.

## Caso de uso 007: Establecimiento de contraseñas por nivel de acceso.

### Actores

- Usuario
- Aplicación

### Descripción

- Esta interfaz muestra y establece contraseñas a los diferentes niveles de acceso en el Router (ruteador).

### Pre-condiciones

- El usuario se encuentra autenticado dentro del sistema.
- El usuario selecciona la pestaña de niveles.

### Post-condiciones

- La contraseña de acceso a un nivel de acceso determinado dentro del Router (ruteador) se ha modificado en la configuración del mismo.

### Flujo normal.

1. La aplicación muestra un campo de captura de la nueva contraseña.
2. La aplicación muestra una tabla con toda la información existente dentro del Router (ruteador).
3. El usuario digita una contraseña.
4. El usuario presiona el botón de añadir.
5. La aplicación añade la tarea a la lista de tareas a ejecutar.
6. Termina caso de uso.

### Flujos alternativos.

4. El usuario no proporcionó una contraseña.
  - La aplicación marca un mensaje de error.
  - La aplicación regresa al paso 3.

Vista.

**Privilegios** **Mensajes** **Auto asegurar** **SNMP** **SSH** **NTP** **Vistas** **ACL** **Auto asegurar 2**  
**Información General** **Contraseña** **Lineas** **Usuarios** **Conexion** **Acceso** **Niveles**

**Nueva contraseña:**

Niveles	Secreto
level 1	habilitado
level 2	deshabilitado
level 3	deshabilitado
level 4	deshabilitado
level 5	habilitado
level 6	deshabilitado
level 7	habilitado
level 8	deshabilitado
level 9	habilitado
level 10	deshabilitado
level 11	deshabilitado
level 12	habilitado
level 13	deshabilitado
level 14	deshabilitado
level 15	deshabilitado

**Añadir**

**Tareas a realizar**  
reconectar  
guardar  
desconectar

**Ejecutar**  **Desconectar**

Notas y asunto:  
Ninguna.

## Caso de uso 008: Nivel de privilegio para ejecutar comandos.

### Actores

- Usuario
- Aplicación

### Descripción

- Esta interfaz muestra el nivel de privilegio mínimo requerido para ejecutar un comando determinado única y exclusivamente si dicho comando no se encuentra el su privilegio establecido por omisión.

### Pre-condiciones

- El usuario se encuentra autenticado dentro de la aplicación.
- El usuario selecciona la pestaña de privilegios.

### Post-condiciones

- El nivel de acceso mínimo para la ejecución de un comando específico dentro de la configuración del Router (ruteador) ha cambiado.

### Flujo normal.

1. La aplicación presenta un cuadro de selección para escoger el nivel.
2. La aplicación muestra un campo de captura para un comando.
3. La aplicación muestra un botón para ver las opciones disponibles del comando.
4. La aplicación muestra un botón para añadir para añadir la opción seleccionada en el campo de opciones disponibles para completar el comando.
5. La aplicación muestra un campo de texto para las opciones posibles para completar el comando.
6. La aplicación muestra la opción "all".
7. La aplicación muestra la opción "reset".
8. La aplicación muestra el botón de añadir.
9. La aplicación muestra una tabla con la información de comandos, niveles y alcance.
10. El usuario selecciona el nivel mínimo para ejecución del comando.
11. El usuario presiona el botón de ver opciones.
12. La aplicación llena el campo de texto con las posibles opciones para completar el comando actual.
13. El usuario selecciona la opción para completar el comando.
14. Si el usuario desea completar todas las opciones posibles del comando va a paso 11 las veces que sean necesarias.
15. El usuario aprieta el botón de añadir opción.
16. La aplicación añade la tarea a la lista de tareas ejecutarse.
17. Termina caso de uso.

### Flujos alternativos.

10. El usuario llena manualmente el campo de comando.
  - Va a paso 15

13. El usuario selecciona la opción “all” para incluir todas las posibles opciones del comando.

- Va a paso 15.

13. El usuario selecciona la opción “reset” para restablecer el comando a su nivel de privilegios original.

- Va a paso 15.

Vista.

The screenshot shows a web-based configuration interface for a network device. The interface has a blue header bar with several tabs: Privilegios, Mensajes, Auto asegurar, SNMP, SSH, NTP, Vistas, ACL, and Auto asegurar 2. Below the header, there are more tabs: Información General, Contraseña, Líneas, Usuarios, Conexión, Acceso, and Niveles. The main content area is divided into several sections. On the left, there is a 'Nivel:' dropdown menu set to '1', a text input field, and two buttons: 'Ver opciones' and 'Añadir opción'. Below these are two checkboxes, 'all' and 'reset', and an 'Añadir' button. On the right, there is a table with three columns: 'Comando', 'Nivel', and 'Alcance'. The table contains three rows: 'ping' with level 5 and scope 'all', 'auto' with level 2, and 'debug' with level 4. At the bottom of the interface, there is a 'Tareas a realizar' section with a list of actions: 'reconectar', 'guardar', and 'desconectar'. Below this list are two buttons: 'Ejecutar' and 'Desconectar', with a text input field between them.

Comando	Nivel	Alcance
ping	5	all
auto	2	
debug	4	

Notas y asunto:

Ninguno.

## Caso de uso 009: Mensajes.

### Actores

- Usuario
- Aplicación

### Descripción

- Esta interfaz configura las opciones de LOG (mensajes) y para ver los mensajes en un tiempo real flexible\*.

Nota: El tiempo real flexible se denomina como aquel en que un pequeño retraso en su ejecución es aceptable.

### Pre-condiciones

- El usuario está autenticado dentro de la aplicación.
- El usuario selecciona la pestaña de mensajes.

### Post-condiciones

- Los cambios dentro de la configuración de LOG (mensajes) se han guardado en el Router (ruteador).

### Flujo normal cambio de configuración.

1. La aplicación muestra un cuadro de selección del tamaño de la terminal de monitoreo.
2. La aplicación muestra un botón de añadir.
3. La aplicación muestra un cuadro de selección para los tipos de mensajes a recibir.
4. La aplicación muestra un botón de añadir.
5. La aplicación muestra un botón de arrancar servicio.
6. La aplicación muestra un botón de determinar.
7. La aplicación muestra una tabla vacía con las columnas de fecha hora y evento.
8. El usuario selecciona el tamaño de la terminal de monitoreo.
9. El usuario presiona el primer botón añadir.
10. La aplicación añade la tarea a la lista de tareas a ejecutarse.
11. El usuario selecciona el tipo de mensajes que desea observar.
12. El usuario presiona el segundo botón de añadir.
13. La aplicación añade la tarea a la lista de tareas a ejecutarse.
14. Termina caso de uso.

### Flujos alternativos cambie de configuración.

- Ninguno.

### Flujo normal monitoreo de mensajes.

1. La aplicación muestra un cuadro de selección del tamaño de la terminal de monitoreo.
2. La aplicación muestra un botón de añadir.
3. La aplicación muestra un cuadro de selección para los tipos de mensajes a recibir.
4. La aplicación muestra un botón de añadir.
5. La aplicación muestra un botón de arrancar servicio.

6. La aplicación muestra un botón de determinar.
7. La aplicación muestra una tabla vacía con las columnas de fecha hora y evento.
8. El usuario presiona el botón de arrancar servicio.
9. La aplicación muestra los mensajes que va recibiendo del Router (ruteador) que va recibiendo en tiempo real flexible.
10. Cuando lo desea el usuario presiona el botón de determinar servicio.
11. Termina caso de uso.

Flujos alternativos cambie de configuración.

- Ninguno.

Vista.

The screenshot shows a software interface for configuring a router. The 'Mensajes' tab is active, displaying various settings and controls. The 'Propiedades' section includes a numeric input for 'Tamaño de la terminal de monitoreo' (4,096) and an 'Añadir' button. The 'Prioridad de mensajes' section features a dropdown menu set to 'default' and another 'Añadir' button. Under 'Ver mensajes', there are 'Arranca servicio' and 'Terminar' buttons. A table with columns 'Fecha', 'Hora', and 'Evento' is present but empty. At the bottom, the 'Tareas a realizar' section lists 'reconectar', 'guardar', and 'desconectar', with 'Ejecutar' and 'Desconectar' buttons below it.

Notas y asunto:

Mientras se tiene encendido el servicio de mensajes no se puede realizar ninguna otra acción hasta que éste se dé por terminado.

## Caso de uso 010: Gestión de vistas.

### Actores

- Usuario
- Aplicación

### Descripción

- Esta interfaz gestiona las vistas existentes dentro del Router (ruteador).

### Pre-condiciones

- El usuario se encuentra autenticado dentro de la aplicación.
- El usuario seleccionó la pestaña de vistas.

### Post-condiciones

- Los cambios sobre las vistas se almacenan en el Router (ruteador)...

### Flujo normal añadir vista.

1. La aplicación muestra un campo de texto para el nombre de la vista.
2. La aplicación muestra un campo de captura para la MIB (base de información de manejo) a incluir.
3. La aplicación muestra dos botones radiales uno con la leyenda "included" y otro con la leyenda "excluded".
4. La aplicación muestra una tabla con el nombre, MIB y la condición.
5. La aplicación muestra un botón de añadir.
6. La aplicación muestra un botón de eliminar.
7. El usuario proporciona el nombre de la vista.
8. El usuario proporciona la MIB.
9. El usuario selecciona el tipo de condición del MIB.
10. El usuario presiona el botón de añadir.
11. La aplicación añade la tarea a la lista de tareas ejecutarse.
12. Termina caso de uso.

### Flujos alternativos añadir vista.

10. El usuario no proporciona un nombre de vista.
  - La aplicación marca un mensaje de error.
  - La aplicación regresa al paso 7.
10. El usuario no proporciona la MIB a incluir.
  - La aplicación marca un mensaje de error.
  - La aplicación regresa al paso 8.

### Flujo normal eliminar vista

1. La aplicación muestra un campo de texto para el nombre de la.
2. La aplicación muestra un campo de captura para la MIB (base de información de manejo) a incluir.
3. La aplicación muestra dos botones radiales uno con la leyenda "included" y otro con la leyenda "excluded".
4. La aplicación muestra una tabla con el nombre, MIB y la condición.
5. La aplicación muestra un botón de añadir.
6. La aplicación muestra un botón de eliminar.
7. El usuario selecciona una de la tabla.

8. El usuario presiona el botón de eliminar.
9. La aplicación añade la tarea a la lista de tareas ejecutarse.
10. Termina caso de uso.

### Flujos alternativos eliminar vista

- Ninguno.

### Vista.

**Vistas**

Nombre de la vista  MIB a incluir   Included  Excluded

Nombre	MIB	Condición
vista	cisco	included
vista	iso.*	included
vista	system	included

**Tareas a realizar**

reconectar  
guardar  
desconectar

### Notas y asunto:

Ninguno.

## Caso de uso 011: Gestión de listas.

### Actores

- Usuario
- Aplicación

### Descripción

- Esta interfaz gestiona las líneas de acceso existentes en el Router (ruteador).

### Pre-condiciones

- Ninguna.

### Post-condiciones

- Se agrega o elimina una lista de la configuración del Router (ruteador)

### Flujo normal añadir lista.

1. La aplicación muestra un campo de selección para el número de lista.
2. La aplicación muestra dos botones radiales con las opciones de “permit” (permitir) y “deny” (denegar).
3. La aplicación muestra un campo de captura del Host (anfitrión) a condicionar.
4. La aplicación muestra una tabla de información con la lista, condición y anfitrión.
5. La aplicación muestra el botón de añadir.
6. La aplicación muestra el botón de eliminar.
7. El usuario selecciona el número de lista a modificar.
8. El usuario selecciona la condición del Host (anfitrión).
9. El usuario proporciona la IP (protocolo de Internet) o el nombre del Host (anfitrión).
10. El usuario presiona el botón de añadir.
11. La aplicación añade la tarea a la lista de tareas ejecutarse.
12. Termina caso de uso.

### Flujos alternativos añadir lista.

10. El usuario no proporcionó un Host (anfitrión).
  - La aplicación marca un mensaje de error.
  - La aplicación regresa al paso 7.

### Flujo normal eliminar lista.

1. La aplicación muestra un campo de selección para el número de lista.
2. La aplicación muestra dos botones radiales con las opciones de “permit” (permitir) y “deny” (denegar).
3. La aplicación muestra un campo de captura del Host (anfitrión) a condicionar.
4. La aplicación muestra una tabla de información con la lista, condición y anfitrión.
5. La aplicación muestra el botón de añadir.
6. La aplicación muestra el botón de eliminar.

7. El usuario selecciona un renglón de la tabla con la información de las listas.
8. El usuario presiona el botón de eliminar.
9. La aplicación añade la tarea a la lista de tareas ejecutarse.
10. Termina caso de uso.

Flujos alternativos eliminar lista.

- Ninguno.

Vista.

ACL  Permit

Número de la lista   Deny Host a integrar en la lista

Lista	Estado	Anfitrión
10	deny	10.1.1.5
10	deny	any log
10	permit	10.1.1.2
10	permit	10.1.1.3
10	permit	10.1.1.4

Tareas a realizar

reconectar  
guardar  
desconectar

Notas y asunto:

Ninguno.

## Caso de uso 012: Configuración de SSH (intérprete de órdenes seguras).

### Actores

- Usuario
- Aplicación

### Descripción

- Esta interfaz permite ver la llave actual en caso de que ésta exista, generar una nueva llave y establecer un anfitrión y dominio para SSH. (intérprete de órdenes seguras).

### Pre-condiciones

- El usuario se encuentra autenticado dentro de la aplicación.
- El usuario seleccionó la pestaña de SSH (intérprete de órdenes segura).

### Post-condiciones

- Se modifica la configuración de SSH (intérprete de órdenes segura) en el Router (ruteador).

### Flujo normal.

1. La aplicación muestra el botón de mostrar llave actual.
2. La aplicación muestra un campo de texto para la llave actual.
3. La aplicación muestra un campo de selección para generar una nueva llave.
4. La aplicación muestra un campo de selección para el tamaño del módulo.
5. La aplicación muestra un campo de captura de texto para la captura del Host (anfitrión).
6. La aplicación muestra un campo de captura de texto para capturar el dominio.
7. La aplicación muestra un botón de añadir.
8. El usuario presiona el botón para mostrar la llave actual.
9. El campo de texto para mostrar la llave actual se muestra en blanco.
10. El usuario selecciona la opción de generar nueva llave.
11. El usuario selecciona el tamaño de módulo que desea para la nueva llave.
12. El usuario proporciona al nombre del Host (anfitrión).
13. El usuario proporciona el dominio.
14. El usuario presiona el botón de añadir.
15. La aplicación añade la tarea a la lista de tareas ejecutarse.
16. Termina caso de uso.

### Flujos alternativos.

9. El campo de texto muestra la llave actual.
  - Va a pasar 10.
14. El usuario no proporciona el nombre del Host (anfitrión).
  - La aplicación muestra un mensaje de error.
  - Va a paso 12.
14. El usuario no proporciona el nombre del dominio).
  - La aplicación muestra un mensaje de error.
  - Va a paso 13.

Vista.

Privilegios Mensajes Auto asegurar SNMP **SSH** NTP Vistas ACL Auto asegurar 2

Información General Contraseña Lineas Usuarios Conexion Acceso Niveles

**Llave RSA**

Mostrar llave actual

Generar nueva llave.

Tamaño de módulo 360

Anfitrión

Dominio

Añadir

Tareas a realizar

reconectar  
guardar  
desconectar

Ejecutar  Desconectar

Notas y asunto:  
Ninguno.

## Caso de uso 013: Creación de comunidades SNMP (protocolo simple de manejo en red).

### Actores

- Usuario
- Aplicación

### Descripción

- Esta interfaz permite añadir y eliminar comunidades SNMP (protocolo simple de manejo en red) de la configuración del Router (ruteador).

### Pre-condiciones

- El usuario se encuentra autenticado dentro de la aplicación.
- El usuario seleccionó la pestaña de SNMP (protocolo simple de manejo en red).

### Post-condiciones

- Se crea una comunidad con las características especificadas dentro de la configuración del Router (ruteador).

### Flujo normal añadir comunidad.

1. La aplicación muestra el campo de captura del nombre de la comunidad.
2. La aplicación muestra el campo de selección del tipo de comunidad.
3. La aplicación muestra un campo de selección con las vistas disponibles dentro del Router (ruteador)...
4. La aplicación muestra un campo de selección con las listas disponibles dentro del Router (ruteador).
5. La aplicación muestra una tabla con la información de las comunidades existentes dentro del Router.
6. La aplicación muestra un botón de añadir.
7. La aplicación muestra un botón de eliminar.
8. El usuario proporciona el nombre de la comunidad.
9. El usuario selecciona el tipo de comunidad que desea crear.
10. El usuario selecciona la vista que desea utilizar para esta comunidad.
11. El usuario selecciona la lista que desea utilizar para esta comunidad.
12. El usuario presiona el botón de añadir.
13. La aplicación añade la tarea a la lista de tareas ejecutarse.
14. Termina caso de uso.

### Flujos alternativos.

10. No se encuentra ninguna vista disponible en el Router (ruteador)...
  - Va a caso de uso 010: gestión de vistas.
10. No se encuentra ninguna lista disponible en el Router (ruteador)...
  - Va a caso de uso 001: gestión de listas.
12. El usuario no proporciona el nombre de la comunidad.
  - La aplicación muestra un mensaje de error.
  - Va a paso 8.

## Vista.

Privilegios Mensajes Auto asegurar SNMP SSH NTP Vistas ACL Auto asegurar 2  
Información General Contraseña Líneas Usuarios Conexión Acceso Niveles

Nombre de la comunidad  tipo de comunidad RO ▼

Comunidades

Vistas disponibles en router.

vista ▼

Listas disponibles en router

10 ▼

Comunidad	Vista	Tipo	ACL
escritura	vista	RW	
lectura	vista	RO	

Añadir Eliminar

Tareas a realizar

reconectar  
guardar  
desconectar

Ejecutar  Desconectar

Notas y asunto:  
Ninguno.

## Caso de uso 014: Gestión de NTP (protocolo de tiempo en red).

### Actores

- Usuario
- Aplicación

### Descripción

- Esta interfaz permite añadir y eliminar servidores NTP (protocolo de tiempo en red) de la configuración del Router.

### Pre-condiciones

- El usuario se encuentra autenticado dentro de la aplicación.
- El usuario seleccionó la pestaña de NTP (protocolo de tiempo en red).

### Post-condiciones

- Se modifica la configuración de los servidores NTP (protocolo de tiempo real) en el Router (ruteador).

### Flujo normal añadir servidor.

1. La aplicación muestra un campo de captura de texto para el nombre o IP (protocolo de Internet) o el nombre del servidor.
2. La aplicación muestra la opción de seleccionar este servidor como preferido.
3. La aplicación muestra el botón de añadir.
4. La aplicación muestra una tabla con la información del tipo de servidor, nombre o IP (protocolo de Internet) y si éste es preferido o no.
5. La aplicación muestra el botón de eliminar.
6. El usuario proporciona el nombre o IP (protocolo de Internet) del servidor.
7. El usuario selecciona la opción de preferido.
8. El usuario presiona el botón de añadir.
9. La aplicación añade la tarea a la lista de tareas a ejecutarse.
10. Termina caso de uso.

### Flujos alternativos añadir servidor.

7. El usuario no selecciona la opción de preferido.
  - Va a paso 8.
8. El usuario no proporciona un nombre o IP (protocolo de Internet) del servidor.
  - La aplicación mandó un mensaje de error.
  - Va a paso 6.

### Flujo normal eliminar servidor.

1. La aplicación muestra un campo de captura de texto para el nombre o IP (protocolo de Internet) o el nombre del servidor.
2. La aplicación muestra la opción de seleccionar este servidor como preferido.
3. La aplicación muestra el botón de añadir.
4. La aplicación muestra una tabla con la información del tipo de servidor, nombre o IP (protocolo de Internet) y si éste es preferido o no.
5. La aplicación muestra el botón de eliminar.
6. El usuario selecciona un renglón de la tabla de información.
7. El usuario presiona el botón de eliminar.
8. Termina caso de uso.

Flujos alternativos eliminar servidor.

- Ninguno.

Vista.

The screenshot shows a window with a blue title bar and standard Windows window controls. The interface has a menu bar with the following items: Privilegios, Mensajes, Auto asegurar, SNMP, SSH, NTP, Vistas, ACL, Auto asegurar 2, Información General, Contraseña, Líneas, Usuarios, Conexión, Acceso, and Niveles. The 'NTP' menu item is highlighted.

Under the 'NTP' menu, there is a section titled 'Tipo de servicio' with a dropdown menu showing 'NTP'. Below this is a text input field for 'Nombre o ip del servidor' and a checkbox labeled 'Preferida'.

A blue 'Añadir' button is positioned above a table. The table has four columns: 'Tipo de servicio', 'Nombre o ip', and 'Preferido'. The first two columns are merged in the header row.

Tipo de servicio		Nombre o ip	Preferido
ntp		10.1.1.3	Sí
ntp		10.1.1.2	Sí

Below the table is a blue 'Eliminar' button.

At the bottom of the window, there is a section titled 'Tareas a realizar' with a list of actions: 'reconectar', 'guardar', and 'desconectar'. Below this list are two buttons: 'Ejecutar' and 'Desconectar', with an empty text input field between them.

Notas y asunto:

Ninguno.

## Caso de uso 015: Auto asegurar.

### Actores

- Usuario
- Aplicación

### Descripción

- Esta interfaz permite ejecutar la instrucción auto asegurar en el Router (ruteador).

### Pre-condiciones

- El usuario se encuentra autenticado dentro de la aplicación.
- El usuario seleccionó la pestaña de auto asegurar.

### Post-condiciones

- Ejecuta el comando de auto asegurar guardándose la configuración dentro del Router y aplicándose a la configuración que se ejecuta actualmente en el Router a petición del usuario.

### Flujo normal.

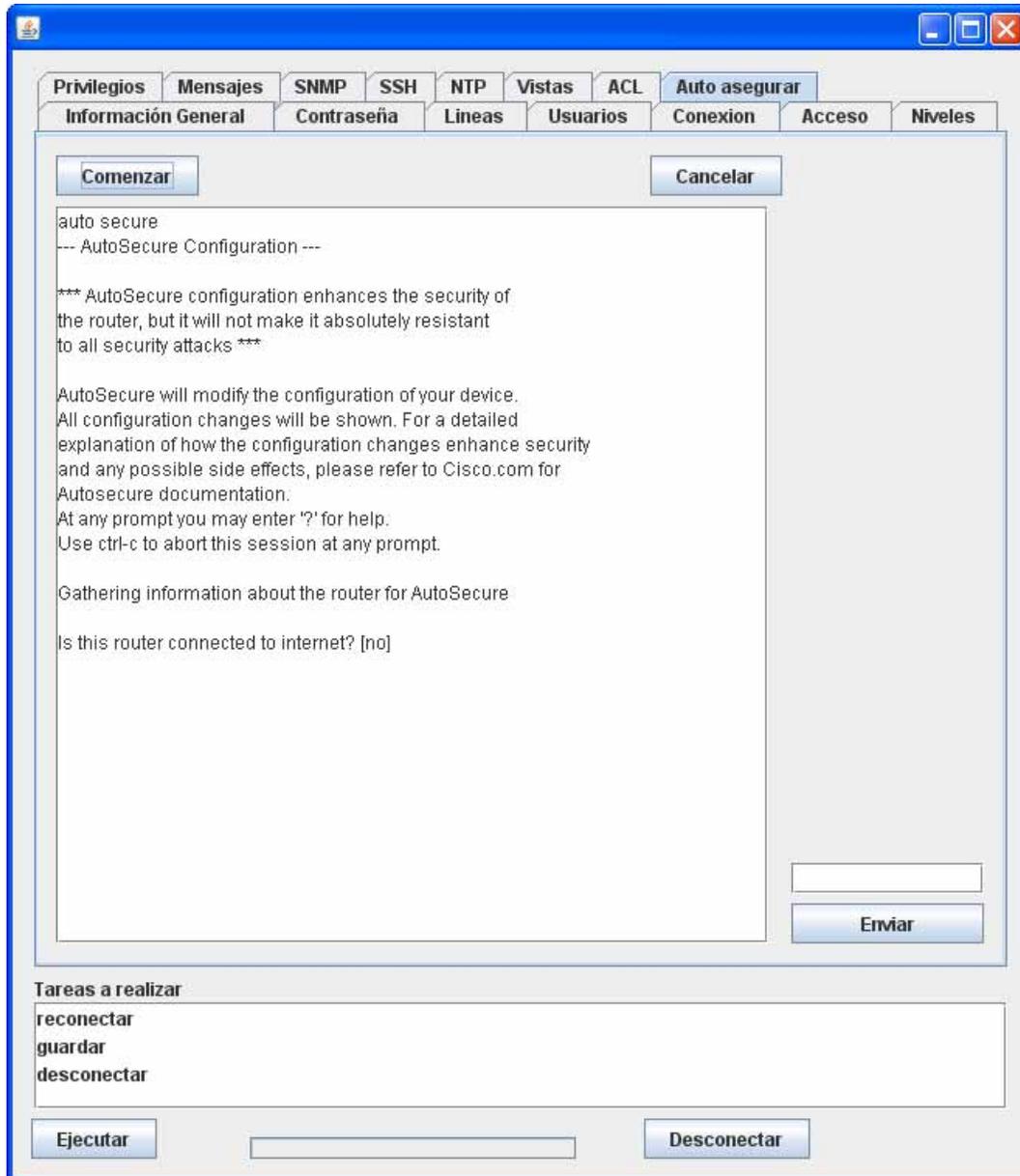
1. La aplicación muestra un botón con la opción de comenzar.
2. La aplicación muestra un botón con opción de cancelar.
3. La aplicación muestra un campo de texto donde se presentarán las preguntas del Router (ruteador).
4. La aplicación muestra un campo de captura de texto para dar respuestas del usuario.
5. La aplicación muestra un botón de enviar.
6. El usuario presiona el botón de comenzar.
7. La aplicación llena el campo de texto con el texto, indicaciones y se detiene al detectar una pregunta provenientes del Router (ruteador).
8. El usuario provee la respuesta del Router en el campo de captura de texto.
9. El usuario presiona la opción de enviar.
10. Mientras el Router presente preguntas va a paso 8, de lo contrario va a paso 11.
11. Termina caso de uso.

### Flujos alternativos.

\*. En cualquier momento entre el paso 6 y el paso 11 el usuario puede presionar el botón de cancelar.

- Termina caso de uso.
9. El usuario no provee una respuesta.
    - La aplicación marca un mensaje de error.
    - La aplicación regresa al paso 8.

Vista.



Notas y asunto:  
Ninguno.

## Caso de uso 016: Ejecución de la lista de tareas.

### Actores

- Usuario
- Aplicación

### Descripción

- En cualquier momento después de terminar cualquiera de los casos de uso del 3 al 14 se pueden ejecutar todas las tareas en la lista de tareas por ejecutarse.

### Pre-condiciones

- El usuario se encuentra autenticado dentro de la aplicación.

### Post-condiciones

- Ejecuta todas y cada una de las tareas en la lista de tareas pendientes de ejecución.
- Limpia la lista de tareas.
- Genera tres nuevas tareas: conectar, salvar y desconectar.

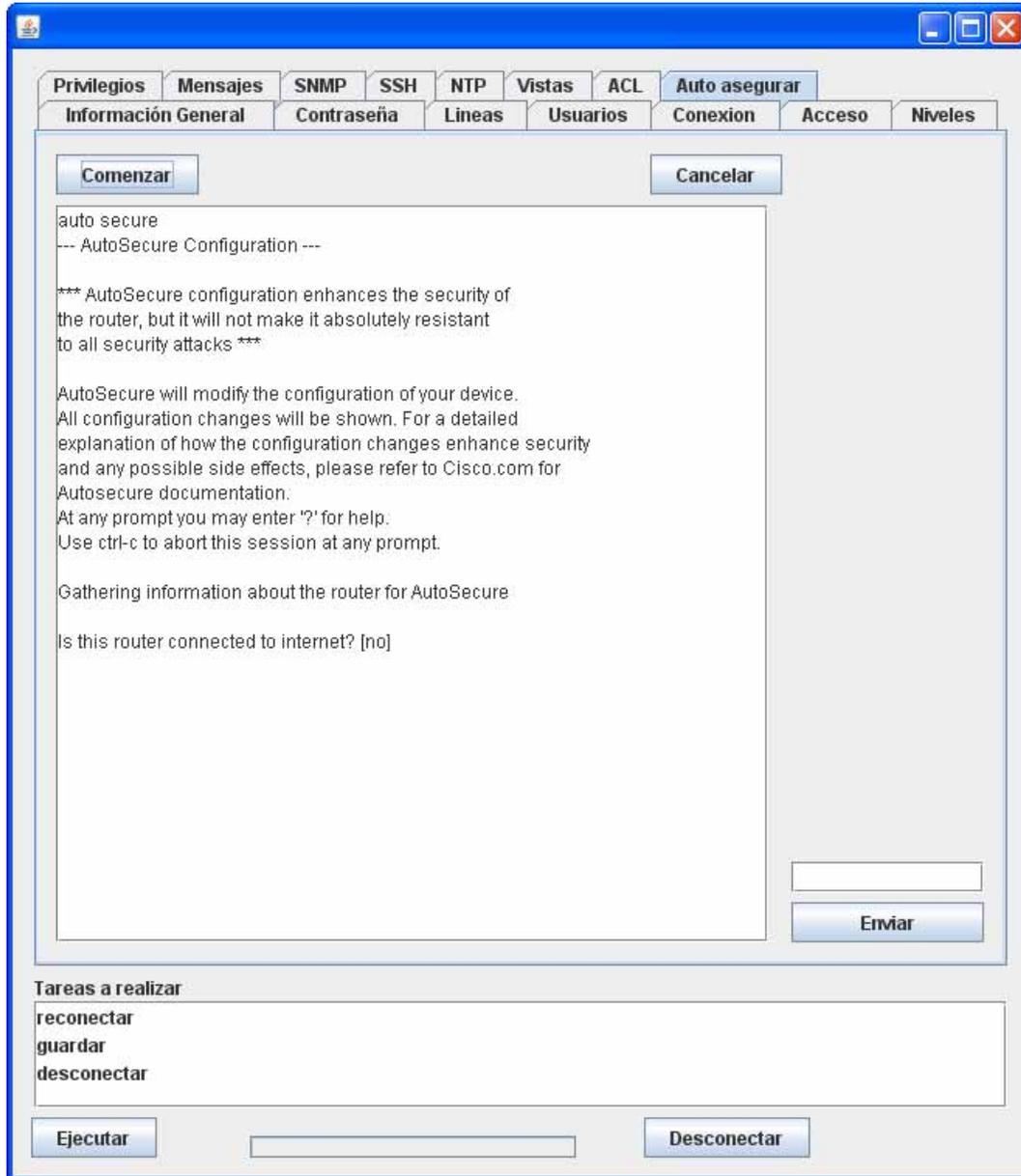
### Flujo normal.

1. La aplicación muestra un campo de texto con tres tareas: conectar, salvar y desconectar.
2. La aplicación muestra un botón de ejecutar.
3. La aplicación muestra un botón de desconectar.
4. El usuario realiza cualquiera de los casos de uso del 3 al 14.
5. La aplicación inserta las nuevas tareas entre las tareas existentes de: conectar y salvar.
6. En cualquier momento después de haber terminado cualquiera de los casos del 3 al 14 el usuario puede presionar el botón de ejecutar.
7. La aplicación ejecuta cada una de las tareas en la lista de tareas a ejecutar.
8. La aplicación actualiza el campo de texto con las tareas poniendo adelante de cada una de las tareas realizadas un indicador de que se realizaron con éxito o no.
9. La aplicación muestra una ventana emergente comunicando que todas las tareas se han realizado con éxito.
10. La aplicación genera tres nuevas tareas: conectar, salvar y desconectar.
11. La aplicación actualiza el campo de texto con las tareas nuevas.
12. Termina caso de uso.

### Flujos alternativos.

- \*. En cualquier momento el usuario puede presionar el botón de desconectar.
  - Se cierra la aplicación.
  - Termina caso de uso.
- 8. Al menos una de las tareas no se realizó con éxito.
  - La aplicación actualiza el campo de texto poniendo al delante de cada una de las tareas que no se realizó con éxito un indicador.
  - La aplicación muestra una ventana emergente indicando cuál fue la primera tarea que no se realizó con éxito.
  - Termina caso de uso.

Vista.



Notas y asunto:

Ninguno.

## ***Código fuente***

## **Código fuente**

### ***Paquete GUI***

### **Clase LoguinGUI**

```
001 /*
002  * To change this template, choose Tools | Templates
003  * and open the template in the editor.
004 */
005
006 /*
007  * LoginGUI.java
008  *
009  * Created on 27/06/2010, 09:37:17 PM
010 */
011 package GUI;
012
013 /**
014  *
015  * @author RAM
016 */
017 import snmplogic.almacenamiento.Data;
018 import snmplogic.tareas.Task;
019 import javax.swing.JFrame;
020 import javax.swing.JOptionPane;
021 import javax.swing.Popup;
022 import snmplogic.*;
023
024 public class LoginGUI extends javax.swing.JFrame {
025
026     Data data = new Data();
027
028     /** Creates new form LoginGUI */
029     public LoginGUI() {
030         initComponents();
031         data = data.getInstance();
032     }
033
034     /** This method is called from within the constructor to
035      * initialize the form.
036      * WARNING: Do NOT modify this code. The content of this method is
037      * always regenerated by the Form Editor.
038      */
039     @SuppressWarnings("unchecked")
040     // <editor-fold defaultstate="collapsed" desc="Generated Code">
041     private void initComponents() {
042
```

```

043     Protocolos = new javax.swing.ButtonGroup();
044     txt_host = new javax.swing.JTextField();
045     btn_host = new javax.swing.JButton();
046     rdb_SSH = new javax.swing.JRadioButton();
047     rdb_Telnet = new javax.swing.JRadioButton();
048     lbl_Ins1 = new javax.swing.JLabel();
049     lbl_ins2 = new javax.swing.JLabel();
050     txt_Usuario = new javax.swing.JTextField();
051     ibi_host = new javax.swing.JLabel();
052     lbl_usuario = new javax.swing.JLabel();
053     lbl_contrasena = new javax.swing.JLabel();
054     btn_Autenticar = new javax.swing.JButton();
055     psw_contrasenia = new javax.swing.JPasswordField();
056     lbl_Ins3 = new javax.swing.JLabel();
057     psw_contraseniaAdmin = new javax.swing.JPasswordField();
058     btn_habilitar = new javax.swing.JButton();
059     lbl_ContraseniaAdmin = new javax.swing.JLabel();
060
061
062     setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
063     txt_host.addActionListener(new java.awt.event.ActionListener() {
064         public void actionPerformed(java.awt.event.ActionEvent evt) {
065             txt_hostActionPerformed(evt);
066         }
067     });
068
069     btn_host.setText("Conectar");
070     btn_host.addActionListener(new java.awt.event.ActionListener() {
071         public void actionPerformed(java.awt.event.ActionEvent evt) {
072             btn_hostActionPerformed(evt);
073         }
074     });
075
076     Protocolos.add(rdb_SSH);
077     rdb_SSH.setText("SSH");
078     rdb_SSH.setEnabled(false);
079     rdb_SSH.addActionListener(new java.awt.event.ActionListener() {
080         public void actionPerformed(java.awt.event.ActionEvent evt) {
081             rdb_SSHActionPerformed(evt);
082         }
083     });
084
085     Protocolos.add(rdb_Telnet);
086     rdb_Telnet.setSelected(true);
087     rdb_Telnet.setText("Telnet");
088
089     lbl_Ins1.setText("1 Escoge tu protocolo (Telnet) y escribe el nombre del
Host");
090

```

```

091     lbl_ins2.setText("2 Proporciona lo siguiente:");
092
093     txt_Usuario.setEditable(false);
094
095     ibi_host.setText("Host");
096
097     lbl_usuario.setText("Usuario");
098
099     lbl_contrasena.setText("Contraseña");
100
101     btn_Autenticar.setText("Autenticar");
102     btn_Autenticar.setEnabled(false);
103     btn_Autenticar.addActionListener(new java.awt.event.ActionListener() {
104         public void actionPerformed(java.awt.event.ActionEvent evt) {
105             btn_AutenticarActionPerformed(evt);
106         }
107     });
108
109     psw_contrasenia.setEnabled(false);
110
111     lbl_Ins3.setText("3 Contraseña de administrador ( enable ):");
112
113     psw_contraseniaAdmin.setEnabled(false);
114
115     btn_habilitar.setText("Habilitar");
116     btn_habilitar.setEnabled(false);
117     btn_habilitar.addActionListener(new java.awt.event.ActionListener() {
118         public void actionPerformed(java.awt.event.ActionEvent evt) {
119             btn_habilitarActionPerformed(evt);
120         }
121     });
122
123     lbl_ContraseniaAdmin.setText("Contraseña");
124
125     javax.swing.GroupLayout layout = new
javax.swing.GroupLayout(getContentPane());
126     getContentPane().setLayout(layout);
127     layout.setHorizontalGroup(
128
layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
129         .addGroup(layout.createSequentialGroup()
130
.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)
131             .addGroup(layout.createSequentialGroup()
layout.createSequentialGroup()
132                 .addContainerGap()
133
.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

```

```

134
.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING
, false)
135         .addGroup(layout.createSequentialGroup()
136             .addGap(10, 10, 10)
137
.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING
)
138         .addComponent(rdb_SSH)
139         .addComponent(rdb_Telnet))
140
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, 69,
Short.MAX_VALUE)
141
.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING
)
142         .addComponent(ibi_host)
143         .addComponent(txt_host,
javax.swing.GroupLayout.Alignment.TRAILING,
javax.swing.GroupLayout.PREFERRED_SIZE, 149,
javax.swing.GroupLayout.PREFERRED_SIZE)))
144         .addComponent(lbl_Ins1))
145         .addGroup(layout.createSequentialGroup()
146
.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING,
false)
147         .addComponent(txt_Usuario,
javax.swing.GroupLayout.Alignment.LEADING,
javax.swing.GroupLayout.DEFAULT_SIZE, 134, Short.MAX_VALUE)
148         .addGroup(layout.createSequentialGroup()
149             .addComponent(lbl_usuario)
150             .addGap(23, 23, 23)))
151
.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING
)
152         .addGroup(layout.createSequentialGroup()
153             .addGap(82, 82, 82)
154             .addComponent(lbl_contrasena))
155         .addGroup(layout.createSequentialGroup()
156             .addGap(65, 65, 65)
157             .addComponent(psw_contrasenia,
javax.swing.GroupLayout.PREFERRED_SIZE, 120,
javax.swing.GroupLayout.PREFERRED_SIZE))))
158         .addComponent(lbl_ins2)))
159         .addGroup(javax.swing.GroupLayout.Alignment.LEADING,
layout.createSequentialGroup()
160             .addGap(96, 96, 96)
161             .addComponent(btn_host))
162         .addGroup(javax.swing.GroupLayout.Alignment.LEADING,
layout.createSequentialGroup()

```

```

163         .addGap(148, 148, 148)
164         .addComponent(btn_Autenticar))
165         .addGroup(javax.swing.GroupLayout.Alignment.LEADING,
layout.createSequentialGroup()
166         .addContainerGap()
167         .addComponent(lbl_Ins3))
168         .addGroup(javax.swing.GroupLayout.Alignment.LEADING,
layout.createSequentialGroup()
169         .addGap(147, 147, 147)
170         .addComponent(btn_habilitar))
171         .addGroup(javax.swing.GroupLayout.Alignment.LEADING,
layout.createSequentialGroup()
172         .addContainerGap()
173         .addComponent(psw_contraseniaAdmin,
javax.swing.GroupLayout.PREFERRED_SIZE, 70,
javax.swing.GroupLayout.PREFERRED_SIZE))
174         .addGroup(javax.swing.GroupLayout.Alignment.LEADING,
layout.createSequentialGroup()
175         .addContainerGap()
176         .addComponent(lbl_ContraseniaAdmin)))
177         .addGap(191, 191, 191))
178     );
179     layout.setVerticalGroup(
180
layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
181         .addGroup(layout.createSequentialGroup()
182         .addContainerGap()
183         .addComponent(lbl_Ins1)
184
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
185
.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING
)
186         .addComponent(rdb_Telnet)
187         .addComponent(ibi_host))
188
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
189
.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELIN
E)
190         .addComponent(rdb_SSH)
191         .addComponent(txt_host,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
192
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
193         .addComponent(btn_host)
194         .addGap(8, 8, 8)
195         .addComponent(lbl_ins2)

```

```

196         .addGap(18, 18, 18)
197
198         .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
199         .addComponent(lbl_usuario)
200         .addComponent(lbl_contrasena))
201
202         .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
203         .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
204         .addComponent(txt_Usuario,
205         javax.swing.GroupLayout.PREFERRED_SIZE,
206         javax.swing.GroupLayout.DEFAULT_SIZE,
207         javax.swing.GroupLayout.PREFERRED_SIZE)
208         .addComponent(psw_contrasenia,
209         javax.swing.GroupLayout.PREFERRED_SIZE,
210         javax.swing.GroupLayout.DEFAULT_SIZE,
211         javax.swing.GroupLayout.PREFERRED_SIZE))
212         .addGap(12, 12, 12)
213         .addComponent(btn_Autenticar)
214
215         .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
216         .addComponent(lbl_Ins3)
217         .addGap(25, 25, 25)
218         .addComponent(lbl_ContraseniaAdmin)
219
220         .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
221         .addComponent(psw_contraseniaAdmin,
222         javax.swing.GroupLayout.PREFERRED_SIZE,
223         javax.swing.GroupLayout.DEFAULT_SIZE,
224         javax.swing.GroupLayout.PREFERRED_SIZE)
225         .addGap(7, 7, 7)
226         .addComponent(btn_habilitar)
227         .addContainerGap()
228     );
229     pack();
230 } // </editor-fold>
231
232 private void btn_hostActionPerformed(java.awt.event.ActionEvent evt) {
233
234     //Establecemos tipo de conexcion en data
235     if (rdb_Telnet.isSelected()) {
236         data.setProtocol(1);
237     } else {
238         data.setProtocol(2);
239     }
240
241     String temp = txt_host.getText();

```

```

230     if (temp.equals("")) {
231         JOptionPane.showMessageDialog(rootPane, "Host vacio");
232     } else {
233         //Creamos tarea y empaquetamos los parametros
234         String parametros[] = {txt_host.getText()};
235         Task login = new Task("login", parametros);
236
237         try {
238
239             if (login.exe()) {
240                 psw_contrasenia.setEnabled(true);
241                 rdb_SSH.setEnabled(false);
242                 rdb_Telnet.setEnabled(false);
243                 txt_host.setEditable(false);
244                 btn_Autenticar.setEnabled(true);
245                 if (data.isAaaEnabled()) {
246                     txt_Usuario.setEditable(true);
247                 } else {
248                     //JOptionPane.showMessageDialog(rootPane, "Usuario o contraseña
incorrecta.");
249                 }
250             }
251         } catch (Exception e) {
252             JOptionPane.showMessageDialog(rootPane, "Error de conexion " + e);
253         }
254     }
255 }
256
257 }
258
259 private void rdb_SSHActionPerformed(java.awt.event.ActionEvent evt) {
260     // TODO add your handling code here:
261 }
262
263 private void txt_hostActionPerformed(java.awt.event.ActionEvent evt) {
264     // TODO add your handling code here:
265 }
266
267 private void btn_AutenticarActionPerformed(java.awt.event.ActionEvent evt) {
268
269     psw_contrasenia.select(0, psw_contrasenia.getPassword().length);
270     String contrsenia = psw_contrasenia.getSelectedText();
271     if (txt_Usuario.isEditable() && txt_Usuario.getText().equals("")) {
272         JOptionPane.showMessageDialog(rootPane, "Usuario vacio.");
273     } else if (psw_contrasenia.isEditable() && contrsenia == null) {
274         JOptionPane.showMessageDialog(rootPane, "Contraseña vacia");
275     } else {
276         try {
277             String parametrosUser[] = {txt_Usuario.getText()};
278             Task sendUser = new Task("sendUser", parametrosUser);

```

```

279
280     psw_contrasenia.select(0, psw_contrasenia.getPassword().length);
281     String parametrosPass[] = {psw_contrasenia.getSelectedText()};
282     Task sendPassword = new Task("sendPassword", parametrosPass);
283
284     boolean userOK = false;
285     boolean passwordOK = false;
286
287     if (data.isAaaEnabled()) {
288         userOK = sendUser.exe();
289         if (userOK) {
290             passwordOK = sendPassword.exe();
291         }
292     } else {
293         passwordOK = sendPassword.exe();
294     }
295
296     if (passwordOK) {
297         txt_Usuario.setEnabled(false);
298         psw_contrasenia.setEditable(false);
299         btn_Autenticar.setEnabled(false);
300
301         psw_contraseniaAdmin.setEnabled(true);
302         btn_habilitar.setEnabled(true);
303
304         btn_host.setEnabled(false);
305
306     } else {
307         JOptionPane.showMessageDialog(rootPane, "Error al autenticarse.");
308     }
309 } catch (Exception e) {
310     JOptionPane.showMessageDialog(rootPane, e);
311 }
312 }
313 }
314
315 private void btn_habilitarActionPerformed(java.awt.event.ActionEvent evt) {
316     psw_contraseniaAdmin.select(0,
psw_contraseniaAdmin.getPassword().length);
317     String contrasenia = psw_contraseniaAdmin.getSelectedText();
318
319     if (contrasenia == null) {
320         JOptionPane.showMessageDialog(rootPane, "Contraseña vacía.");
321     } else {
322         String parametros[] = {"15", psw_contraseniaAdmin.getSelectedText()};
323         Task enableAccess = new Task("enableAccess", parametros);
324
325         try {
326             if (enableAccess.exe()) {
327                 LoginGUI.this.setVisible(false);

```

```

328         LoginGUI.this.dispose();
329         Task desconectar = new Task("Desconectar");
330
331         desconectar.exe();
332         new MainGUI().setVisible(true);
333
334         LoginGUI.this.setVisible(false);
335         LoginGUI.this.dispose();
336
337     } else {
338         JOptionPane.showMessageDialog(rootPane, "Error al habilitar
administrador");
339     }
340 } catch (Exception e) {
341     JOptionPane.showMessageDialog(rootPane, e);
342 }
343 }
344 }
345
346 /**
347  * @param args the command line arguments
348  */
349 public static void main(String args[]) {
350     java.awt.EventQueue.invokeLater(new Runnable() {
351
352         public void run() {
353             new LoginGUI().setVisible(true);
354         }
355     });
356 }
357 // Variables declaration - do not modify
358 private javax.swing.ButtonGroup Protocolos;
359 private javax.swing.JButton btn_Autenticar;
360 private javax.swing.JButton btn_habilitar;
361 private javax.swing.JButton btn_host;
362 private javax.swing.JLabel ibi_host;
363 private javax.swing.JLabel lbl_ContraseniaAdmin;
364 private javax.swing.JLabel lbl_Ins1;
365 private javax.swing.JLabel lbl_Ins3;
366 private javax.swing.JLabel lbl_contrasena;
367 private javax.swing.JLabel lbl_ins2;
368 private javax.swing.JLabel lbl_usuario;
369 private javax.swing.JPasswordField psw_contrasenia;
370 private javax.swing.JPasswordField psw_contraseniaAdmin;
371 private javax.swing.JRadioButton rdb_SSH;
372 private javax.swing.JRadioButton rdb_Telnet;
373 private javax.swing.JTextField txt_Usuario;
374 private javax.swing.JTextField txt_host;
375 // End of variables declaration
376 }

```

## Clase MainGUI

```
0001 /*
0002  * To change this template, choose Tools | Templates
0003  * and open the template in the editor.
0004 */
0005
0006 /*
0007  * any.java
0008  *
0009  * Created on 30/06/2010, 07:38:27 PM
0010 */
0011 package GUI;
0012
0013 import java.awt.Color;
0014 import java.util.ArrayList;
0015 import java.util.Iterator;
0016 import java.util.List;
0017 //import javax.swing.ComboBoxModel;
0018 import java.util.Observable;
0019 import java.util.Observer;
0020 import java.util.regex.Matcher;
0021 import java.util.regex.Pattern;
0022 import javax.swing.DefaultComboBoxModel;
0023 import javax.swing.DefaultListModel;
0024 //import javax.swing.JFrame;
0025 import javax.swing.JOptionPane;
0026 //import javax.swing.ListModel;
0027 import javax.swing.ListSelectionModel;
0028 import javax.swing.table.DefaultTableModel;
0029 import snmplogic.almacenamiento.Data;
0030 import snmplogic.tareas.ShowRunInfo;
0031 import snmplogic.tareas.Task;
0032
0033 /**
0034  *
0035  * @author RAM
0036  */
0037 public class MainGUI extends javax.swing.JFrame implements Observer {
0038
0039     //public static final char STX = (char) 0x02;
0040     public static final char ETX = (char) 0x03;
0041     Data data = new Data();
0042     List listaDeTareas = new ArrayList<Task>();
0043     ShowRunInfo showRunInfo = new ShowRunInfo();
0044     DefaultListModel listaModelo = new DefaultListModel();
0045     int listaModeloExe = 0;
0046     int autoSecureStep = 0;
```

```

0047 String bufferTemp = "";
0048 int numeroDeTareas = 0;
0049
0050 /** Creates new form any */
0051 public MainGUI() {
0052     initComponents();
0053     //Informacion general
0054     data = data.getInstance();
0055     IG_lbl_HostName.setText(data.getHostName());
0056     //Vistas
0057
0058     load();
0059
0060 }
0061
0062 /** This method is called from within the constructor to
0063     * initialize the form.
0064     * WARNING: Do NOT modify this code. The content of this method is
0065     * always regenerated by the Form Editor.
0066     */
0067 @SuppressWarnings("unchecked")
0068 // <editor-fold defaultstate="collapsed" desc="Generated Code">
0069 private void initComponents() {
0070
0071     SCR_btn_habilitar = new javax.swing.ButtonGroup();
0072     PRI_btngrp_opciones = new javax.swing.ButtonGroup();
0073     AS_btngrp_internet = new javax.swing.ButtonGroup();
0074     AS_btngrp_SNMP = new javax.swing.ButtonGroup();
0075     AS_btngrp_SSH = new javax.swing.ButtonGroup();
0076     AS_btngrp_firewall = new javax.swing.ButtonGroup();
0077     AS_btngrp_aplicar = new javax.swing.ButtonGroup();
0078     vis_btngrp_includedExcluded = new javax.swing.ButtonGroup();
0079     acl_btngrp_acl = new javax.swing.ButtonGroup();
0080     tab_principal = new javax.swing.JTabbedPane();
0081     pan_InformacionGeneral = new javax.swing.JPanel();
0082     IG_lbl_dispositivo = new javax.swing.JLabel();
0083     IG_scr_showTechSupport = new javax.swing.JScrollPane();
0084     IG_txta_showTechSupport = new javax.swing.JTextArea();
0085     IG_lbl_HostName = new javax.swing.JLabel();
0086     btn_ShowTechSupport = new javax.swing.JButton();
0087     pan_secret = new javax.swing.JPanel();
0088     SCR_lbl_secret = new javax.swing.JLabel();
0089     SCT_btn_aniadir = new javax.swing.JButton();
0090     SCR_txt_SecretPassword = new javax.swing.JTextField();
0091     SCR_lbl_habilitado = new javax.swing.JLabel();
0092     SCR_lbl_texto = new javax.swing.JLabel();
0093     SCR_rbtn_habilitar = new javax.swing.JRadioButton();
0094     SCR_rbtn_deshabilitar = new javax.swing.JRadioButton();
0095     SCR_lbl_solicitud = new javax.swing.JLabel();
0096     pan_lineas = new javax.swing.JPanel();

```

```
0097 LIN_cbox_linea = new javax.swing.JComboBox();
0098 LIN_lbl_lineas = new javax.swing.JLabel();
0099 LIN_lbl_contrasenia = new javax.swing.JLabel();
0100 LIN_txt_nContrasenia = new javax.swing.JTextField();
0101 LIN_scrpnl_lineas = new javax.swing.JScrollPane();
0102 LIN_tbl_lineas = new javax.swing.JTable();
0103 LIN_btn_aniadir = new javax.swing.JButton();
0104 pan_usuarios = new javax.swing.JPanel();
0105 US_scrpnl_usuarios = new javax.swing.JScrollPane();
0106 US_tbl_usuarios = new javax.swing.JTable();
0107 US_btn_eliminar = new javax.swing.JButton();
0108 US_txt_usuario = new javax.swing.JTextField();
0109 US_lbl_usuario = new javax.swing.JLabel();
0110 US_lbl_nivelPrivilegios = new javax.swing.JLabel();
0111 US_sld_privilegios = new javax.swing.JSlider();
0112 US_cbox_encryptacion = new javax.swing.JComboBox();
0113 US_lbl_encryptacion = new javax.swing.JLabel();
0114 US_txt_contrasenia = new javax.swing.JTextField();
0115 US_lbl_contrasenia = new javax.swing.JLabel();
0116 US_btn_aniadir = new javax.swing.JButton();
0117 US_lbl_noPrivilegio = new javax.swing.JLabel();
0118 pan_conexion = new javax.swing.JPanel();
0119 CON_lbl_linea = new javax.swing.JLabel();
0120 CON_cbox_lineas = new javax.swing.JComboBox();
0121 CON_lbl_minutos = new javax.swing.JLabel();
0122 CON_scrpnl_conexiones = new javax.swing.JScrollPane();
0123 CON_tbl_tiempos = new javax.swing.JTable();
0124 CON_bytn_aniadir = new javax.swing.JButton();
0125 CON_spin_min = new javax.swing.JSpinner();
0126 CON_spin_sec = new javax.swing.JSpinner();
0127 CON_lbl_segundos = new javax.swing.JLabel();
0128 pan_acceso = new javax.swing.JPanel();
0129 ACC_lbl_intentos = new javax.swing.JLabel();
0130 ACC_spn_intentos = new javax.swing.JSpinner();
0131 ACC_btn_aniadir = new javax.swing.JButton();
0132 ACC_txt_numeros = new javax.swing.JTextField();
0133 pan_niveles = new javax.swing.JPanel();
0134 NIV_scrpnl_niveles = new javax.swing.JScrollPane();
0135 NIV_tbl_niveles = new javax.swing.JTable();
0136 NIV_lbl_contrasenia = new javax.swing.JLabel();
0137 NIV_txt_nContrasenia = new javax.swing.JTextField();
0138 NIV_btn_aniadir = new javax.swing.JButton();
0139 pan_privilegios = new javax.swing.JPanel();
0140 PRI_scrpnl_privilegios = new javax.swing.JScrollPane();
0141 PRI_tbl_privilegios = new javax.swing.JTable();
0142 PRI_lbl_nivel = new javax.swing.JLabel();
0143 PRI_spn_nivel = new javax.swing.JSpinner();
0144 PRI_txt_comando = new javax.swing.JTextField();
0145 jScrollPane1 = new javax.swing.JScrollPane();
0146 PRI_lst_opciones = new javax.swing.JList();
```

```
0147 PRI_btn_opciones = new javax.swing.JButton();
0148 jButton1 = new javax.swing.JButton();
0149 PRI_btn_aniadir = new javax.swing.JButton();
0150 PRI_chkbtn_all = new javax.swing.JCheckBox();
0151 PRI_chkbtn_reset = new javax.swing.JCheckBox();
0152 pan_mensajes = new javax.swing.JPanel();
0153 MNS_lbl_terminal = new javax.swing.JLabel();
0154 MNS_spn_terminal = new javax.swing.JSpinner();
0155 MNS_lbl_prioridad = new javax.swing.JLabel();
0156 MNS_cbox_prioridad = new javax.swing.JComboBox();
0157 MNS_btn_prioridad = new javax.swing.JButton();
0158 MNS_btn_comenzar = new javax.swing.JButton();
0159 MNS_btn_terminar = new javax.swing.JButton();
0160 MNS_scrpnl_mensajes = new javax.swing.JScrollPane();
0161 MNS_tbl_mensajes = new javax.swing.JTable();
0162 MNS_lbl_propiedades = new javax.swing.JLabel();
0163 MNS_lbl_ver = new javax.swing.JLabel();
0164 MNS_btn_terminal = new javax.swing.JButton();
0165 snmp_lbl_comunidades = new javax.swing.JPanel();
0166 snmp_lbl_vistas = new javax.swing.JLabel();
0167 snmp_cbox_vistas = new javax.swing.JComboBox();
0168 snmp_lbl_listas = new javax.swing.JLabel();
0169 snmp_cbox_listas = new javax.swing.JComboBox();
0170 snmp_lbl_titulo = new javax.swing.JLabel();
0171 snmp_lbl_nombre = new javax.swing.JLabel();
0172 snmp_txtf_nombre = new javax.swing.JTextField();
0173 snmp_lbl_tipo = new javax.swing.JLabel();
0174 snmp_cbox_tipo = new javax.swing.JComboBox();
0175 snmp_btn_aniadir = new javax.swing.JButton();
0176 snmp_scrpnl_info = new javax.swing.JScrollPane();
0177 snmp_tbl_info = new javax.swing.JTable();
0178 snmp_btn_eliminar = new javax.swing.JButton();
0179 pan_ssh = new javax.swing.JPanel();
0180 ssh_lbl_RSA = new javax.swing.JLabel();
0181 ssh_lbl_modulo = new javax.swing.JLabel();
0182 ssh_spn_tamano = new javax.swing.JSpinner();
0183 ssh_scrpnl_llaves = new javax.swing.JScrollPane();
0184 ssh_txta_llaves = new javax.swing.JTextArea();
0185 ssh_lbl_anfitrión = new javax.swing.JLabel();
0186 ssh_txtf_anfitrión = new javax.swing.JTextField();
0187 ssh_lbl_dominio = new javax.swing.JLabel();
0188 ssh_txtf_dominio = new javax.swing.JTextField();
0189 ssh_btn_aniadir = new javax.swing.JButton();
0190 ssh_check_nuevaLlave = new javax.swing.JCheckBox();
0191 jButton2 = new javax.swing.JButton();
0192 pan_ntp = new javax.swing.JPanel();
0193 ntp_lbl_tipoServicio = new javax.swing.JLabel();
0194 ntp_cbox_tipoServicio = new javax.swing.JComboBox();
0195 ntp_lbl_nombreServicio = new javax.swing.JLabel();
0196 ntp_txtf_nombreServicio = new javax.swing.JTextField();
```

```
0197 ntp_check_prefered = new javax.swing.JCheckBox();
0198 ntp_check_key = new javax.swing.JCheckBox();
0199 ntp_txtf_numeroLlave = new javax.swing.JTextField();
0200 ntp_lbl_numeroLlave = new javax.swing.JLabel();
0201 ntp_lbl_valorLlave = new javax.swing.JLabel();
0202 ntp_txtf_valorLlave = new javax.swing.JTextField();
0203 ntp_btn_aniadir = new javax.swing.JButton();
0204 ntp_scrpnl_servidores = new javax.swing.JScrollPane();
0205 ntp_tbl_servidores = new javax.swing.JTable();
0206 ntp_btn_eliminar = new javax.swing.JButton();
0207 pan_vistas = new javax.swing.JPanel();
0208 vis_lbl_nombre = new javax.swing.JLabel();
0209 vis_txtf_MIB = new javax.swing.JTextField();
0210 vis_btn_eliminar = new javax.swing.JButton();
0211 vis_txtf_nombre = new javax.swing.JTextField();
0212 vis_btn_aniadir = new javax.swing.JButton();
0213 jScrollPane10 = new javax.swing.JScrollPane();
0214 vis_tbl_info = new javax.swing.JTable();
0215 vis_lbl_vistas = new javax.swing.JLabel();
0216 vis_lbl_mib = new javax.swing.JLabel();
0217 vis_btn_included = new javax.swing.JRadioButton();
0218 vis_btn_excluded = new javax.swing.JRadioButton();
0219 pan_acl = new javax.swing.JPanel();
0220 acl_scrpnl_info = new javax.swing.JScrollPane();
0221 acl_tbl_info = new javax.swing.JTable();
0222 acl_lbl_host = new javax.swing.JLabel();
0223 acl_spn_numerolista = new javax.swing.JSpinner();
0224 acl_lbl_numerolista = new javax.swing.JLabel();
0225 acl_lbl_acl = new javax.swing.JLabel();
0226 acl_txtf_host = new javax.swing.JTextField();
0227 acl_btn_aniadir = new javax.swing.JButton();
0228 acl_btn_eliminar = new javax.swing.JButton();
0229 acl_btn_permit = new javax.swing.JRadioButton();
0230 acl_btn_deny = new javax.swing.JRadioButton();
0231 pan_auto2 = new javax.swing.JPanel();
0232 aaut_btn_comenzar = new javax.swing.JButton();
0233 aaut_scrpnl_info = new javax.swing.JScrollPane();
0234 aauto_txta_info = new javax.swing.JTextArea();
0235 aaut_btn_enviar = new javax.swing.JButton();
0236 aaut_txtf_message = new javax.swing.JTextField();
0237 jButton3 = new javax.swing.JButton();
0238 lbl_tareas = new javax.swing.JLabel();
0239 btn_ejecutar = new javax.swing.JButton();
0240 pgrb_ejecutar = new javax.swing.JProgressBar();
0241 btn_desconectar = new javax.swing.JButton();
0242 jScrollPane2 = new javax.swing.JScrollPane();
0243 lst_tareas = new javax.swing.JList();
0244
0245
setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
```

```

0246
0247     IG_lbl_dispositivo.setText("Dispositivo:");
0248
0249     IG_txta_showTechSupport.setColumns(20);
0250     IG_txta_showTechSupport.setRows(5);
0251     IG_scr_showTechSupport.setViewportView(IG_txta_showTechSupport);
0252
0253     btn_ShowTechSupport.setText("Mostrar información");
0254     btn_ShowTechSupport.addActionListener(new
java.awt.event.ActionListener() {
0255         public void actionPerformed(java.awt.event.ActionEvent evt) {
0256             btn_ShowTechSupportActionPerformed(evt);
0257         }
0258     });
0259
0260     javax.swing.GroupLayout pan_InformacionGeneralLayout = new
javax.swing.GroupLayout(pan_InformacionGeneral);
0261     pan_InformacionGeneral.setLayout(pan_InformacionGeneralLayout);
0262     pan_InformacionGeneralLayout.setHorizontalGroup(
0263 pan_InformacionGeneralLayout.createParallelGroup(javax.swing.GroupLayout.Alignm
ent.LEADING)
0264         .addGroup(pan_InformacionGeneralLayout.createSequentialGroup()
0265             .addGap(10, 10, 10)
0266             .addGroup(pan_InformacionGeneralLayout.createParallelGroup(javax.s
wing.GroupLayout.Alignment.LEADING)
0267                 .addComponent(IG_scr_showTechSupport,
javax.swing.GroupLayout.DEFAULT_SIZE, 602, Short.MAX_VALUE)
0268                 .addGroup(pan_InformacionGeneralLayout.createSequentialGroup()
0269                     .addComponent(IG_lbl_dispositivo)
0270                     .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.R
ELATED)
0271                     .addComponent(IG_lbl_HostName,
javax.swing.GroupLayout.PREFERRED_SIZE, 115,
javax.swing.GroupLayout.PREFERRED_SIZE))
0272                 .addComponent(btn_ShowTechSupport))
0273             .addGap(10, 10, 10)
0274         );
0275     pan_InformacionGeneralLayout.setVerticalGroup(
0276 pan_InformacionGeneralLayout.createParallelGroup(javax.swing.GroupLayout.Alignm
ent.LEADING)
0277         .addGroup(pan_InformacionGeneralLayout.createSequentialGroup()
0278             .addGap(10, 10, 10)
0279             .addGroup(pan_InformacionGeneralLayout.createParallelGroup(javax.s
wing.GroupLayout.Alignment.BASELINE)
0280                 .addComponent(IG_lbl_dispositivo)

```

```

0281         .addComponent(IG_lbl_HostName,
javax.swing.GroupLayout.PREFERRED_SIZE, 14,
javax.swing.GroupLayout.PREFERRED_SIZE))
0282
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
0283         .addComponent(IG_scr_showTechSupport,
javax.swing.GroupLayout.DEFAULT_SIZE, 465, Short.MAX_VALUE)
0284
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
0285         .addComponent(btn_ShowTechSupport)
0286         .addContainerGap()
0287     );
0288
0289     tab_principal.addTab("Información General", pan_InformacionGeneral);
0290
0291     SCR_lbl_secret.setText("Password");
0292
0293     SCT_btn_aniadir.setText("Añadir");
0294     SCT_btn_aniadir.addActionListener(new java.awt.event.ActionListener() {
0295         public void actionPerformed(java.awt.event.ActionEvent evt) {
0296             SCT_btn_aniadirActionPerformed(evt);
0297         }
0298     });
0299
0300     SCR_lbl_habilitado.setText("<secreto habilitado?>");
0301
0302     SCR_lbl_texto.setText("Si deseas establecer un password nuevo:");
0303
0304     SCR_btn_habilitar.add(SCR_rbtn_habilitar);
0305     SCR_rbtn_habilitar.setSelected(true);
0306     SCR_rbtn_habilitar.setText("Habilitar");
0307     SCR_rbtn_habilitar.addActionListener(new java.awt.event.ActionListener()
{
0308         public void actionPerformed(java.awt.event.ActionEvent evt) {
0309             SCR_rbtn_habilitarActionPerformed(evt);
0310         }
0311     });
0312
0313     SCR_btn_habilitar.add(SCR_rbtn_deshabilitar);
0314     SCR_rbtn_deshabilitar.setText("Deshabilitar");
0315     SCR_rbtn_deshabilitar.addActionListener(new
java.awt.event.ActionListener() {
0316         public void actionPerformed(java.awt.event.ActionEvent evt) {
0317             SCR_rbtn_deshabilitarActionPerformed(evt);
0318         }
0319     });
0320
0321     SCR_lbl_solicitud.setText("-");
0322

```

```

0323     javax.swing.GroupLayout pan_secretLayout = new
javax.swing.GroupLayout(pan_secret);
0324     pan_secret.setLayout(pan_secretLayout);
0325     pan_secretLayout.setHorizontalGroup(
0326
pan_secretLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING
)
0327         .addGroup(pan_secretLayout.createSequentialGroup()
0328             .addContainerGap()
0329
.addGroup(pan_secretLayout.createParallelGroup(javax.swing.GroupLayout.Alignment
.LEADING)
0330             .addGroup(pan_secretLayout.createSequentialGroup()
0331                 .addComponent(SCR_lbl_secret)
0332
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
0333                 .addComponent(SCR_lbl_habilitado))
0334                 .addComponent(SCR_lbl_texto)
0335                 .addComponent(SCR_rbtn_habilita
0336                 .addComponent(SCT_btn_aniadir)
0337                 .addComponent(SCR_lbl_solicitud)
0338
.addGroup(pan_secretLayout.createParallelGroup(javax.swing.GroupLayout.Alignment
.TRAILING, false)
0339                 .addComponent(SCR_txt_SecretPassword,
javax.swing.GroupLayout.Alignment.LEADING)
0340                 .addComponent(SCR_rbtn_deshabilita
javax.swing.GroupLayout.Alignment.LEADING,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)))
0341             .addContainerGap(410, Short.MAX_VALUE))
0342         );
0343     pan_secretLayout.setVerticalGroup(
0344
pan_secretLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING
)
0345         .addGroup(pan_secretLayout.createSequentialGroup()
0346             .addContainerGap()
0347
.addGroup(pan_secretLayout.createParallelGroup(javax.swing.GroupLayout.Alignment
.BASELINE)
0348                 .addComponent(SCR_lbl_secret)
0349                 .addComponent(SCR_lbl_habilitado))
0350
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
0351                 .addComponent(SCR_lbl_texto)
0352
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
0353                 .addComponent(SCR_rbtn_habilita

```

```

0354 .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
0355     .addComponent(SCR_rbtn_deshabilitar)
0356     .addGap(18, 18, 18)
0357     .addComponent(SCR_lbl_solicitud)
0358     .addGap(16, 16, 16)
0359     .addComponent(SCR_txt_SecretPassword,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
0360
0361 .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
0362     .addComponent(SCT_btn_aniadir)
0363     .addContainerGap(346, Short.MAX_VALUE))
0364 );
0365 tab_principal.addTab("Contraseña", pan_secret);
0366
0367 LIN_lbl_lineas.setText("Linea");
0368
0369 LIN_lbl_contrasenia.setText("Nueva contraseña");
0370
0371 LIN_tbl_lineas.setModel(new javax.swing.table.DefaultTableModel(
0372     new Object [][] {
0373         {null, null, null, null},
0374         {null, null, null, null},
0375         {null, null, null, null},
0376         {null, null, null, null}
0377     },
0378     new String [] {
0379         "Title 1", "Title 2", "Title 3", "Title 4"
0380     }
0381 ));
0382 LIN_scrpnl_lineas.setViewportView(LIN_tbl_lineas);
0383
0384 LIN_btn_aniadir.setText("Añadir");
0385 LIN_btn_aniadir.addActionListener(new java.awt.event.ActionListener() {
0386     public void actionPerformed(java.awt.event.ActionEvent evt) {
0387         LIN_btn_aniadirActionPerformed(evt);
0388     }
0389 });
0390
0391 javax.swing.GroupLayout pan_lineasLayout = new
javax.swing.GroupLayout(pan_lineas);
0392 pan_lineas.setLayout(pan_lineasLayout);
0393 pan_lineasLayout.setHorizontalGroup(
0394
pan_lineasLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING
)
0395     .addGroup(pan_lineasLayout.createSequentialGroup()

```

```

0396         .addContainerGap()
0397
0398         .addGroup(pan_lineasLayout.createParallelGroup(javax.swing.GroupLayout.Alignment
.LEADING)
0399
0400         .addComponent(LIN_lbl_contrasenia)
0401
0402         .addGroup(pan_lineasLayout.createParallelGroup(javax.swing.GroupLayout.Alignment
.TRAILING, false)
0403         .addComponent(LIN_txt_nContrasenia,
javax.swing.GroupLayout.Alignment.LEADING)
0404         .addComponent(LIN_cbox_linea,
javax.swing.GroupLayout.Alignment.LEADING, 0, 110, Short.MAX_VALUE)
0405         .addComponent(LIN_lbl_lineas,
javax.swing.GroupLayout.Alignment.LEADING))
0406         .addComponent(LIN_btn_aniadir)
0407         .addGap(31, 31, 31)
0408         .addComponent(LIN_scrpnl_lineas,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
0409         .addGap(259, 259, 259))
0410     );
0411     pan_lineasLayout.setVerticalGroup(
0412     pan_lineasLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING
)
0413     .addGroup(pan_lineasLayout.createSequentialGroup()
0414     .addComponent(LIN_scrpnl_lineas,
javax.swing.GroupLayout.PREFERRED_SIZE, 329,
javax.swing.GroupLayout.PREFERRED_SIZE)
0415     .addContainerGap()
0416     .addGroup(pan_lineasLayout.createSequentialGroup()
0417     .addComponent(LIN_lbl_lineas)
0418     .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
0419     .addComponent(LIN_cbox_linea,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
0420     .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
0421     .addComponent(LIN_lbl_contrasenia)
0422     .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

```

```

0423         .addComponent(LIN_txt_nContrasenia,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
0424
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, 57,
Short.MAX_VALUE)
0425         .addComponent(LIN_btn_aniadir)
0426         .addContainerGap(354, Short.MAX_VALUE))))
0427     );
0428
0429     tab_principal.addTab("Lineas", pan_lineas);
0430
0431     US_tbl_usuarios.setModel(new javax.swing.table.DefaultTableModel(
0432         new Object [][] {
0433             {null, null, null, null},
0434             {null, null, null, null},
0435             {null, null, null, null},
0436             {null, null, null, null}
0437         },
0438         new String [] {
0439             "Title 1", "Title 2", "Title 3", "Title 4"
0440         }
0441     ));
0442     US_scrpnl_usuarios.setViewportView(US_tbl_usuarios);
0443
0444     US_btn_eliminar.setText("Eliminar");
0445     US_btn_eliminar.addActionListener(new java.awt.event.ActionListener() {
0446         public void actionPerformed(java.awt.event.ActionEvent evt) {
0447             US_btn_eliminarActionPerformed(evt);
0448         }
0449     });
0450
0451     US_lbl_usuario.setText("Usuario");
0452
0453     US_lbl_nivelPrivilegios.setText("Nivel de privilegios");
0454
0455     US_sld_privilegios.setMajorTickSpacing(1);
0456     US_sld_privilegios.setMaximum(15);
0457     US_sld_privilegios.setMinimum(1);
0458     US_sld_privilegios.setMinorTickSpacing(1);
0459     US_sld_privilegios.setPaintTicks(true);
0460     US_sld_privilegios.setSnapToTicks(true);
0461     US_sld_privilegios.setValue(1);
0462     US_sld_privilegios.addChangeListener(new
javax.swing.event.ChangeListener() {
0463         public void stateChanged(javax.swing.event.ChangeEvent evt) {
0464             US_sld_privilegiosStateChanged(evt);
0465         }
0466     });

```

```

0467
0468     US_cbox_encryption.setModel(new
javax.swing.DefaultComboBoxModel(new String[] { "password", "secret" }));
0469
0470     US_lbl_encryption.setText("Encriptación");
0471
0472     US_lbl_contraseña.setText("Contraseña");
0473
0474     US_btn_anadir.setText("Añadir");
0475     US_btn_anadir.addActionListener(new java.awt.event.ActionListener() {
0476         public void actionPerformed(java.awt.event.ActionEvent evt) {
0477             US_btn_anadirActionPerformed(evt);
0478         }
0479     });
0480
0481     US_lbl_noPrivilegio.setText("1");
0482
0483     javax.swing.GroupLayout pan_usuariosLayout = new
javax.swing.GroupLayout(pan_usuarios);
0484     pan_usuarios.setLayout(pan_usuariosLayout);
0485     pan_usuariosLayout.setHorizontalGroup(
0486     pan_usuariosLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
0487         .addGroup(pan_usuariosLayout.createSequentialGroup()
0488             .addContainerGap()
0489             .addGroup(pan_usuariosLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
0490                 .addComponent(US_btn_eliminar)
0491                 .addComponent(US_btn_anadir)
0492                 .addComponent(US_sld_privilegios,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
0493                 .addGroup(pan_usuariosLayout.createSequentialGroup()
0494                     .addGroup(pan_usuariosLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING, false)
0495                         .addGroup(javax.swing.GroupLayout.Alignment.LEADING,
pan_usuariosLayout.createSequentialGroup()
0496                             .addComponent(US_lbl_nivelPrivilegios)
0497                             .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
0498                             .addComponent(US_lbl_noPrivilegio,
javax.swing.GroupLayout.PREFERRED_SIZE, 44,
javax.swing.GroupLayout.PREFERRED_SIZE))
0499                         .addGroup(javax.swing.GroupLayout.Alignment.LEADING,
pan_usuariosLayout.createSequentialGroup()

```

```

0500
.addGroup(pan_usuariosLayout.createParallelGroup(javax.swing.GroupLayout.Alignm
ent.TRAILING)
0501             .addComponent(US_txt_usuario,
javax.swing.GroupLayout.Alignment.LEADING,
javax.swing.GroupLayout.PREFERRED_SIZE, 118,
javax.swing.GroupLayout.PREFERRED_SIZE)
0502             .addComponent(US_lbl_usuario,
javax.swing.GroupLayout.Alignment.LEADING))
0503             .addGap(77, 77, 77)
0504
.addGroup(pan_usuariosLayout.createParallelGroup(javax.swing.GroupLayout.Alignm
ent.LEADING)
0505             .addComponent(US_cbox_encryptacion,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
0506             .addComponent(US_lbl_encryptacion))))
0507             .addGap(60, 60, 60)
0508
.addGroup(pan_usuariosLayout.createParallelGroup(javax.swing.GroupLayout.Alignm
ent.LEADING)
0509             .addComponent(US_lbl_contrasenia)
0510             .addComponent(US_txt_contrasenia,
javax.swing.GroupLayout.PREFERRED_SIZE, 124,
javax.swing.GroupLayout.PREFERRED_SIZE)))
0511             .addComponent(US_scrpnl_usuarios,
javax.swing.GroupLayout.PREFERRED_SIZE, 588,
javax.swing.GroupLayout.PREFERRED_SIZE))
0512             .addContainerGap(24, Short.MAX_VALUE))
0513 );
0514 pan_usuariosLayout.setVerticalGroup(
0515
pan_usuariosLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADI
NG)
0516             .addGroup(pan_usuariosLayout.createSequentialGroup()
0517             .addComponent(US_scrpnl_usuarios,
javax.swing.GroupLayout.PREFERRED_SIZE, 155,
javax.swing.GroupLayout.PREFERRED_SIZE)
0518             .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
0519             .addComponent(US_btn_eliminar)
0520             .addGap(8, 8, 8)
0521             .addGroup(pan_usuariosLayout.createParallelGroup(javax.swing.GroupLayout.Alignm
ent.BASELINE)
0522             .addComponent(US_lbl_usuario)
0523             .addComponent(US_lbl_encryptacion)
0524             .addComponent(US_lbl_contrasenia))

```

```

0525 .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
0526 .addGroup(pan_usuariosLayout.createParallelGroup(javax.swing.GroupLayout.Alignm
ent.BASELINE)
0527     .addComponent(US_txt_usuario,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
0528     .addComponent(US_cbox_encryption,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
0529     .addComponent(US_txt_contrasenia,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
0530     .addGap(18, 18, 18)
0531 .addGroup(pan_usuariosLayout.createParallelGroup(javax.swing.GroupLayout.Alignm
ent.BASELINE)
0532     .addComponent(US_lbl_nivelPrivilegios)
0533     .addComponent(US_lbl_noPrivilegio))
0534 .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
0535     .addComponent(US_sld_privilegios,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
0536 .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, 189,
Short.MAX_VALUE)
0537     .addComponent(US_btn_aniadir)
0538     .addContainerGap()
0539 );
0540
0541 tab_principal.addTab("Usuarios", pan_usuarios);
0542
0543 CON_lbl_linea.setText("Linea");
0544
0545 CON_lbl_minutos.setText("Minutos");
0546
0547 CON_tbl_tiempos.setModel(new javax.swing.table.DefaultTableModel(
0548     new Object [][] {
0549         {null, null, null, null},
0550         {null, null, null, null},
0551         {null, null, null, null},
0552         {null, null, null, null}
0553     },
0554     new String [] {

```

```

0555         "Title 1", "Title 2", "Title 3", "Title 4"
0556     }
0557 ));
0558 CON_scrpnl_conexiones.setViewportView(CON_tbl_tiempos);
0559
0560 CON_bytn_aniadir.setText("Añadir");
0561 CON_bytn_aniadir.addActionListener(new java.awt.event.ActionListener() {
0562     public void actionPerformed(java.awt.event.ActionEvent evt) {
0563         CON_bytn_aniadirActionPerformed(evt);
0564     }
0565 });
0566
0567 CON_spin_min.setModel(new javax.swing.SpinnerNumberModel(0, 0, 59,
0568 1));
0569 CON_spin_sec.setModel(new javax.swing.SpinnerNumberModel(0, 0, 59,
0570 1));
0571 CON_lbl_segundos.setText("Segundos");
0572
0573 javax.swing.GroupLayout pan_conexionLayout = new
0574 javax.swing.GroupLayout(pan_conexion);
0575 pan_conexion.setLayout(pan_conexionLayout);
0576 pan_conexionLayout.setHorizontalGroup(
0577     pan_conexionLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
0578     .addGroup(pan_conexionLayout.createSequentialGroup()
0579     .addGroup(pan_conexionLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
0580     .addGroup(pan_conexionLayout.createSequentialGroup()
0581     .addComponent(CON_scrpnl_conexiones,
0582     javax.swing.GroupLayout.PREFERRED_SIZE, 586,
0583     javax.swing.GroupLayout.PREFERRED_SIZE)
0584     .addGap(10, 10, 10)
0585     .addGroup(pan_conexionLayout.createSequentialGroup()
0586     .addGroup(pan_conexionLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
0587     .addComponent(CON_cbox_lineas, 0, 94,
0588     javax.swing.GroupLayout.PREFERRED_SIZE)
0589     .addComponent(CON_lbl_linea)

```

```

0589
.addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
pan_conexionLayout.createSequentialGroup()
0590
.addGroup(pan_conexionLayout.createParallelGroup(javax.swing.GroupLayout.Alignm
ent.LEADING)
0591
.addComponent(CON_spin_min,
javax.swing.GroupLayout.PREFERRED_SIZE, 41,
javax.swing.GroupLayout.PREFERRED_SIZE)
0592
.addComponent(CON_lbl_minutos))
0593
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
0594
.addGroup(pan_conexionLayout.createParallelGroup(javax.swing.GroupLayout.Alignm
ent.LEADING)
0595
.addComponent(CON_lbl_segundos)
0596
.addComponent(CON_spin_sec,
javax.swing.GroupLayout.PREFERRED_SIZE, 42,
javax.swing.GroupLayout.PREFERRED_SIZE))))
0597
.addGap(31, 31, 31))
0598
.addComponent(CON_bytn_aniadir))
0599
.addGap(625, 625, 625))))
0600 );
0601 pan_conexionLayout.setVerticalGroup(
0602
pan_conexionLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADI
NG)
0603
.addGroup(pan_conexionLayout.createSequentialGroup()
0604
.addContainerGap()
0605
.addComponent(CON_lbl_linea)
0606
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
0607
.addComponent(CON_cbox_lineas,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
0608
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
0609
.addGroup(pan_conexionLayout.createParallelGroup(javax.swing.GroupLayout.Alignm
ent.BASELINE)
0610
.addComponent(CON_lbl_minutos)
0611
.addComponent(CON_lbl_segundos))
0612
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
0613
.addGroup(pan_conexionLayout.createParallelGroup(javax.swing.GroupLayout.Alignm
ent.BASELINE)

```

```

0614         .addComponent(CON_spin_min,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
0615         .addComponent(CON_spin_sec,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
0616         .addGap(18, 18, 18)
0617         .addComponent(CON_bytn_aniadir)
0618
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
0619         .addComponent(CON_scrpnl_conexiones,
javax.swing.GroupLayout.PREFERRED_SIZE, 329,
javax.swing.GroupLayout.PREFERRED_SIZE)
0620         .addContainerGap(58, Short.MAX_VALUE))
0621     );
0622
0623     tab_principal.addTab("Conexion", pan_conexion);
0624
0625     pan_acceso.setEnabled(false);
0626
0627     ACC_lbl_intentos.setText("Número de intentos establecidos:");
0628
0629     ACC_spn_intentos.setModel(new javax.swing.SpinnerNumberModel(2, 2,
1024, 1));
0630
0631     ACC_btn_aniadir.setText("Añadir");
0632     ACC_btn_aniadir.addActionListener(new java.awt.event.ActionListener() {
0633         public void actionPerformed(java.awt.event.ActionEvent evt) {
0634             ACC_btn_aniadirActionPerformed(evt);
0635         }
0636     });
0637
0638     ACC_txt_numeros.setText("#");
0639     ACC_txt_numeros.setEnabled(false);
0640     ACC_txt_numeros.addActionListener(new java.awt.event.ActionListener() {
0641         public void actionPerformed(java.awt.event.ActionEvent evt) {
0642             ACC_txt_numerosActionPerformed(evt);
0643         }
0644     });
0645
0646     javax.swing.GroupLayout pan_accesoLayout = new
javax.swing.GroupLayout(pan_acceso);
0647     pan_acceso.setLayout(pan_accesoLayout);
0648     pan_accesoLayout.setHorizontalGroup(
0649
pan_accesoLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
0650         .addGroup(pan_accesoLayout.createSequentialGroup()

```

```

0651         .addContainerGap()
0652
.addGroup(pan_accesoLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
0653         .addComponent(ACC_btn_aniadir)
0654         .addGroup(pan_accesoLayout.createSequentialGroup()
0655         .addComponent(ACC_lbl_intentos)
0656
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
0657         .addComponent(ACC_txt_numeros,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
0658         .addComponent(ACC_spn_intentos,
javax.swing.GroupLayout.PREFERRED_SIZE, 38,
javax.swing.GroupLayout.PREFERRED_SIZE))
0659         .addContainerGap(434, Short.MAX_VALUE))
0660     );
0661     pan_accesoLayout.setVerticalGroup(
0662     pan_accesoLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
0663     .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
pan_accesoLayout.createSequentialGroup()
0664     .addContainerGap()
0665
.addGroup(pan_accesoLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
0666     .addComponent(ACC_lbl_intentos)
0667     .addComponent(ACC_txt_numeros,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
0668
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
0669     .addComponent(ACC_spn_intentos,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
0670
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, 440,
Short.MAX_VALUE)
0671     .addComponent(ACC_btn_aniadir)
0672     .addContainerGap()
0673     );
0674
0675     tab_principal.addTab("Acceso", pan_acceso);
0676
0677     NIV_tbl_niveles.setModel(new javax.swing.table.DefaultTableModel(
0678     new Object [][] {

```

```

0679         {null, null, null, null},
0680         {null, null, null, null},
0681         {null, null, null, null},
0682         {null, null, null, null}
0683     },
0684     new String [] {
0685         "Title 1", "Title 2", "Title 3", "Title 4"
0686     }
0687 ));
0688 NIV_scrpnl_niveles.setViewportView(NIV_tbl_niveles);
0689
0690 NIV_lbl_contrasenia.setText("Nueva contraseña:");
0691
0692 NIV_btn_aniadir.setText("Añadir");
0693 NIV_btn_aniadir.addActionListener(new java.awt.event.ActionListener() {
0694     public void actionPerformed(java.awt.event.ActionEvent evt) {
0695         NIV_btn_aniadirActionPerformed(evt);
0696     }
0697 });
0698
0699 javax.swing.GroupLayout pan_nivelesLayout = new
javax.swing.GroupLayout(pan_niveles);
0700 pan_niveles.setLayout(pan_nivelesLayout);
0701 pan_nivelesLayout.setHorizontalGroup(
0702 pan_nivelesLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
0703     .addGroup(pan_nivelesLayout.createSequentialGroup()
0704         .addGap(10, 10, 10)
0705         .addGroup(pan_nivelesLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
0706             .addComponent(NIV_scrpnl_niveles,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
0707             .addGroup(pan_nivelesLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)
0708                 .addComponent(NIV_lbl_contrasenia,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
0709                 .addComponent(NIV_btn_aniadir)
0710                 .addComponent(NIV_txt_nContrasenia)))
0711             .addGap(10, 10, 10)
0712         );
0713 pan_nivelesLayout.setVerticalGroup(
0714 pan_nivelesLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
0715     .addGroup(pan_nivelesLayout.createSequentialGroup()
0716         .addGap(10, 10, 10)

```

```

0717         .addComponent(NIV_lbl_contrasenia)
0718
0719         .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
0719         .addComponent(NIV_txt_nContrasenia,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
0720         .addGap(18, 18, 18)
0721         .addComponent(NIV_scrpnl_niveles,
javax.swing.GroupLayout.PREFERRED_SIZE, 315,
javax.swing.GroupLayout.PREFERRED_SIZE)
0722
0722         .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, 118,
Short.MAX_VALUE)
0723         .addComponent(NIV_btn_aniadir)
0724         .addContainerGap()
0725     );
0726
0727     tab_principal.addTab("Niveles", pan_niveles);
0728
0729     PRI_tbl_privilegios.setModel(new javax.swing.table.DefaultTableModel(
0730     new Object [][] {
0731         {null, null, null, null},
0732         {null, null, null, null},
0733         {null, null, null, null},
0734         {null, null, null, null}
0735     },
0736     new String [] {
0737         "Title 1", "Title 2", "Title 3", "Title 4"
0738     }
0739     ));
0740     PRI_scrpnl_privilegios.setViewportViewView(PRI_tbl_privilegios);
0741
0742     PRI_lbl_nivel.setText("Nivel:");
0743
0744     PRI_spn_nivel.setModel(new javax.swing.SpinnerNumberModel(1, 1, 15,
1));
0745
0746
0747     PRI_lst_opciones.setSelectionMode(javax.swing.ListSelectionModel.SINGLE_SELEC
TION);
0747     jScrollPane1.setViewportViewView(PRI_lst_opciones);
0748
0749     PRI_btn_opciones.setText("Ver opciones");
0750     PRI_btn_opciones.addActionListener(new java.awt.event.ActionListener() {
0751     public void actionPerformed(java.awt.event.ActionEvent evt) {
0752         PRI_btn_opcionesActionPerformed(evt);
0753     }
0754     });
0755

```

```

0756     jButton1.setText("Añadir opcion");
0757     jButton1.addActionListener(new java.awt.event.ActionListener() {
0758         public void actionPerformed(java.awt.event.ActionEvent evt) {
0759             jButton1ActionPerformed(evt);
0760         }
0761     });
0762
0763     PRI_btn_aniadir.setText("Añadir");
0764     PRI_btn_aniadir.addActionListener(new java.awt.event.ActionListener() {
0765         public void actionPerformed(java.awt.event.ActionEvent evt) {
0766             PRI_btn_aniadirActionPerformed(evt);
0767         }
0768     });
0769
0770     PRI_btngrp_opciones.add(PRI_chkbtn_all);
0771     PRI_chkbtn_all.setText("all");
0772     PRI_chkbtn_all.addActionListener(new java.awt.event.ActionListener() {
0773         public void actionPerformed(java.awt.event.ActionEvent evt) {
0774             PRI_chkbtn_allActionPerformed(evt);
0775         }
0776     });
0777
0778     PRI_btngrp_opciones.add(PRI_chkbtn_reset);
0779     PRI_chkbtn_reset.setText("reset");
0780     PRI_chkbtn_reset.addActionListener(new java.awt.event.ActionListener() {
0781         public void actionPerformed(java.awt.event.ActionEvent evt) {
0782             PRI_chkbtn_resetActionPerformed(evt);
0783         }
0784     });
0785
0786     javax.swing.GroupLayout pan_privilegiosLayout = new
javax.swing.GroupLayout(pan_privilegios);
0787     pan_privilegios.setLayout(pan_privilegiosLayout);
0788     pan_privilegiosLayout.setHorizontalGroup(
0789     pan_privilegiosLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
0790         .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
pan_privilegiosLayout.createSequentialGroup()
0791             .addComponent(jScrollPane1,
javax.swing.GroupLayout.PREFERRED_SIZE, 153,
javax.swing.GroupLayout.PREFERRED_SIZE)
0792             .addGroup(pan_privilegiosLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
0793                 .addGroup(pan_privilegiosLayout.createSequentialGroup()
0794                     .addComponent(jScrollPane1,
javax.swing.GroupLayout.PREFERRED_SIZE, 153,
javax.swing.GroupLayout.PREFERRED_SIZE)
0795                     .addGroup(pan_privilegiosLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

```

```

0796         .addGroup(pan_privilegiosLayout.createSequentialGroup())
0797         .addGap(37, 37, 37)
0798
0799         .addGroup(pan_privilegiosLayout.createParallelGroup(javax.swing.GroupLayout.Align
ment.LEADING)
0800         .addComponent(PRI_chkbtn_all)
0801         .addComponent(PRI_chkbtn_reset)))
0802         .addGroup(pan_privilegiosLayout.createSequentialGroup())
0803         .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
0804         .addComponent(PRI_btn_aniadir)))
0805         .addGroup(pan_privilegiosLayout.createSequentialGroup())
0806         .addComponent(PRI_lbl_nivel)
0807         .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
0808         .addComponent(PRI_spn_nivel,
javax.swing.GroupLayout.PREFERRED_SIZE, 40,
javax.swing.GroupLayout.PREFERRED_SIZE))
0809         .addGroup(pan_privilegiosLayout.createParallelGroup(javax.swing.GroupLayout.Align
ment.TRAILING, false)
0810         .addComponent(PRI_txt_comando,
javax.swing.GroupLayout.Alignment.LEADING)
0811         .addGroup(javax.swing.GroupLayout.Alignment.LEADING,
pan_privilegiosLayout.createSequentialGroup())
0812         .addComponent(PRI_btn_opciones)
0813         .addGap(18, 18, 18)
0814         .addComponent(jButton1)))
0815         .addGap(18, 18, 18)
0816         .addComponent(PRI_scrpnl_privilegios,
javax.swing.GroupLayout.PREFERRED_SIZE, 325,
javax.swing.GroupLayout.PREFERRED_SIZE)
0817         .addGap(127, 127, 127))
0818         );
0819         pan_privilegiosLayout.setVerticalGroup(
pan_privilegiosLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEA
DING)
0820         .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
pan_privilegiosLayout.createSequentialGroup())
0821         .addGroup(pan_privilegiosLayout.createParallelGroup(javax.swing.GroupLayout.Align
ment.TRAILING)
0822         .addComponent(PRI_scrpnl_privilegios,
javax.swing.GroupLayout.Alignment.LEADING,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
0823         .addGroup(pan_privilegiosLayout.createSequentialGroup())
0824         .addContainerGap())

```

```

0825
.addGroup(pan_privilegiosLayout.createParallelGroup(javax.swing.GroupLayout.Align
ment.BASELINE)
0826         .addComponent(PRI_lbl_nivel)
0827         .addComponent(PRI_spn_nivel,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
0828
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
0829         .addComponent(PRI_txt_comando,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
0830         .addGap(10, 10, 10)
0831
.addGroup(pan_privilegiosLayout.createParallelGroup(javax.swing.GroupLayout.Align
ment.BASELINE)
0832         .addComponent(PRI_btn_opciones)
0833         .addComponent(jButton1))
0834
.addGroup(pan_privilegiosLayout.createParallelGroup(javax.swing.GroupLayout.Align
ment.LEADING)
0835         .addGroup(pan_privilegiosLayout.createSequentialGroup())
0836         .addGap(18, 18, 18)
0837         .addComponent(jScrollPane1,
javax.swing.GroupLayout.DEFAULT_SIZE, 417, Short.MAX_VALUE))
0838         .addGroup(pan_privilegiosLayout.createSequentialGroup())
0839         .addGap(36, 36, 36)
0840         .addComponent(PRI_chkbtn_all)
0841
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
0842         .addComponent(PRI_chkbtn_reset)
0843
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
0844         .addComponent(PRI_btn_aniadir))))
0845     .addContainerGap()
0846 );
0847
0848 tab_principal.addTab("Privilegios", pan_privilegios);
0849
0850 MNS_lbl_terminal.setText("Tamaño de la terminal de monitoreo");
0851
0852 MNS_spn_terminal.setModel(new javax.swing.SpinnerNumberModel(4096,
4096, 2147483647, 1));
0853
0854 MNS_lbl_prioridad.setText("Prioridad de mensajes");
0855
0856 MNS_cbox_prioridad.setModel(new
javax.swing.DefaultComboBoxModel(new String[] { "default", "emergencias

```

```

(severity=0)", "alerts (severity=1)", "critical (severity=2)", "errors (severity=3)",
"warnings (severity=4)", "notifications (severity=5)", "informational (severity=6)",
"debugging (severity=7)", " ", " ", " ", " ", " ", " ", " " ));
0857
0858     MNS_btn_prioridad.setText("Añadir");
0859     MNS_btn_prioridad.addActionListener(new java.awt.event.ActionListener()
{
0860         public void actionPerformed(java.awt.event.ActionEvent evt) {
0861             MNS_btn_prioridadActionPerformed(evt);
0862         }
0863     });
0864
0865     MNS_btn_comenzar.setText("Arranca servicio");
0866     MNS_btn_comenzar.addActionListener(new java.awt.event.ActionListener()
{
0867         public void actionPerformed(java.awt.event.ActionEvent evt) {
0868             MNS_btn_comenzarActionPerformed(evt);
0869         }
0870     });
0871
0872     MNS_btn_terminar.setText("Terminar");
0873     MNS_btn_terminar.addActionListener(new java.awt.event.ActionListener()
{
0874         public void actionPerformed(java.awt.event.ActionEvent evt) {
0875             MNS_btn_terminarActionPerformed(evt);
0876         }
0877     });
0878
0879     MNS_tbl_mensajes.setModel(new javax.swing.table.DefaultTableModel(
0880         new Object [][] {
0881             {null, null, null, null},
0882             {null, null, null, null},
0883             {null, null, null, null},
0884             {null, null, null, null}
0885         },
0886         new String [] {
0887             "Title 1", "Title 2", "Title 3", "Title 4"
0888         }
0889     ));
0890     MNS_scrpnl_mensajes.setViewportView(MNS_tbl_mensajes);
0891
0892     MNS_lbl_propiedades.setText("Propiedades");
0893
0894     MNS_lbl_ver.setText("Ver mensajes");
0895
0896     MNS_btn_terminal.setText("Añadir");
0897     MNS_btn_terminal.addActionListener(new java.awt.event.ActionListener() {
0898         public void actionPerformed(java.awt.event.ActionEvent evt) {
0899             MNS_btn_terminalActionPerformed(evt);
0900         }

```

```

0901     });
0902
0903     javax.swing.GroupLayout pan_mensajesLayout = new
javax.swing.GroupLayout(pan_mensajes);
0904     pan_mensajes.setLayout(pan_mensajesLayout);
0905     pan_mensajesLayout.setHorizontalGroup(
0906
pan_mensajesLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
0907         .addGroup(pan_mensajesLayout.createSequentialGroup()
0908             .addContainerGap()
0909
.addGroup(pan_mensajesLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
0910             .addComponent(MNS_scrpnl_mensajes,
javax.swing.GroupLayout.PREFERRED_SIZE, 584,
javax.swing.GroupLayout.PREFERRED_SIZE)
0911             .addComponent(MNS_lbl_propiedades)
0912             .addComponent(MNS_btn_terminal)
0913             .addComponent(MNS_btn_prioridad)
0914             .addComponent(MNS_lbl_ver)
0915             .addGroup(pan_mensajesLayout.createSequentialGroup()
0916
.addGroup(pan_mensajesLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
0917             .addComponent(MNS_lbl_terminal)
0918             .addComponent(MNS_lbl_prioridad)
0919             .addComponent(MNS_btn_comenzar))
0920             .addGap(66, 66, 66)
0921
.addGroup(pan_mensajesLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
0922             .addComponent(MNS_btn_terminar)
0923
.addGroup(pan_mensajesLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)
0924             .addComponent(MNS_cbox_prioridad, 0, 0,
Short.MAX_VALUE)
0925             .addComponent(MNS_spn_terminal))))))
0926             .addContainerGap(28, Short.MAX_VALUE))
0927     );
0928     pan_mensajesLayout.setVerticalGroup(
0929
pan_mensajesLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
0930         .addGroup(pan_mensajesLayout.createSequentialGroup()
0931             .addContainerGap()
0932             .addComponent(MNS_lbl_propiedades)
0933
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

```

```

0934
.addGroup(pan_mensajesLayout.createParallelGroup(javax.swing.GroupLayout.Alignm
ent.BASELINE)
0935         .addComponent(MNS_spn_terminal,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
0936         .addComponent(MNS_lbl_terminal))
0937
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
0938         .addComponent(MNS_btn_terminal)
0939         .addGap(66, 66, 66)
0940
.addGroup(pan_mensajesLayout.createParallelGroup(javax.swing.GroupLayout.Alignm
ent.BASELINE)
0941         .addComponent(MNS_lbl_prioridad)
0942         .addComponent(MNS_cbox_prioridad,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
0943
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
0944         .addComponent(MNS_btn_prioridad)
0945
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
0946         .addComponent(MNS_lbl_ver)
0947
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
0948
.addGroup(pan_mensajesLayout.createParallelGroup(javax.swing.GroupLayout.Alignm
ent.BASELINE)
0949         .addComponent(MNS_btn_comenzar)
0950         .addComponent(MNS_btn_terminar))
0951
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
0952         .addComponent(MNS_scrpnl_mensajes,
javax.swing.GroupLayout.PREFERRED_SIZE, 488,
javax.swing.GroupLayout.PREFERRED_SIZE)
0953         .addGap(55, 55, 55))
0954     );
0955
0956     tab_principal.addTab("Mensajes", pan_mensajes);
0957
0958     snmp_lbl_vistas.setText("Vistas disponibles en router.");
0959
0960     snmp_cbox_vistas.setModel(new javax.swing.DefaultComboBoxModel(new
String[] { "Vista 1", "vista 2", "vista etc" }));
0961
0962     snmp_lbl_listas.setText("Listas disponibles en router");
0963

```

```

0964     snmp_cbox_listas.setModel(new javax.swing.DefaultComboBoxModel(new
String[] { "Lista 1", "Lista 5", "Lista etc" }));
0965
0966     snmp_lbl_titulo.setText("Comunidades");
0967
0968     snmp_lbl_nombre.setText("Nombre de la comunidad");
0969
0970     snmp_lbl_tipo.setText("tipo de comunidad");
0971
0972     snmp_cbox_tipo.setModel(new javax.swing.DefaultComboBoxModel(new
String[] { "RO", "RW" }));
0973
0974     snmp_btn_aniadir.setText("Añadir");
0975     snmp_btn_aniadir.addActionListener(new java.awt.event.ActionListener() {
0976         public void actionPerformed(java.awt.event.ActionEvent evt) {
0977             snmp_btn_aniadirActionPerformed(evt);
0978         }
0979     });
0980
0981     snmp_tbl_info.setModel(new javax.swing.table.DefaultTableModel(
0982         new Object [][] {
0983             {null, null, null, null},
0984             {null, null, null, null},
0985             {null, null, null, null},
0986             {null, null, null, null}
0987         },
0988         new String [] {
0989             "Title 1", "Title 2", "Title 3", "Title 4"
0990         }
0991     ));
0992     snmp_scrpnl_info.setViewportView(snmp_tbl_info);
0993
0994     snmp_btn_eliminar.setText("Eliminar");
0995     snmp_btn_eliminar.addActionListener(new java.awt.event.ActionListener() {
0996         public void actionPerformed(java.awt.event.ActionEvent evt) {
0997             snmp_btn_eliminarActionPerformed(evt);
0998         }
0999     });
1000
1001     javax.swing.GroupLayout snmp_lbl_comunidadesLayout = new
javax.swing.GroupLayout(snmp_lbl_comunidades);
1002     snmp_lbl_comunidades.setLayout(snmp_lbl_comunidadesLayout);
1003     snmp_lbl_comunidadesLayout.setHorizontalGroup(
1004     snmp_lbl_comunidadesLayout.createParallelGroup(javax.swing.GroupLayout.Alignme
nt.LEADING)
1005         .addGroup(snmp_lbl_comunidadesLayout.createSequentialGroup()
1006             .addContainerGap()

```

```

1007
.addGroup(snmplblcomunidadesLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
1008     .addComponent(snmplscrplnlnfo,
javax.swing.GroupLayout.DEFAULT_SIZE, 602, Short.MAX_VALUE)
1009     .addGroup(snmplblcomunidadesLayout.createSequentialGroup())
1010     .addComponent(snmpllblnombre)
1011
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
1012     .addComponent(snmpltxtfnombre,
javax.swing.GroupLayout.PREFERRED_SIZE, 90,
javax.swing.GroupLayout.PREFERRED_SIZE)
1013
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
1014     .addComponent(snmpllbltipo)
1015
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
1016     .addComponent(snmplcboxtipo,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
1017     .addComponent(snmpllbltitulo)
1018     .addComponent(snmpllblvistas,
javax.swing.GroupLayout.DEFAULT_SIZE, 602, Short.MAX_VALUE)
1019     .addComponent(snmplcboxvistas,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
1020     .addComponent(snmpllbllistas)
1021     .addComponent(snmplcboxlistas,
javax.swing.GroupLayout.PREFERRED_SIZE, 118,
javax.swing.GroupLayout.PREFERRED_SIZE)
1022     .addGroup(snmplblcomunidadesLayout.createSequentialGroup())
1023     .addComponent(snmplbtnaniadir)
1024
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
1025     .addComponent(snmplbtneliminar)))
1026     .addContainerGap()
1027 );
1028 snmplblcomunidadesLayout.setVerticalGroup(
1029 snmplblcomunidadesLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
1030     .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
snmplblcomunidadesLayout.createSequentialGroup())
1031     .addContainerGap()
1032
.addGroup(snmplblcomunidadesLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

```

```

1033         .addComponent(snmptxtf_nombre,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
1034         .addComponent(snmplbl_tipo)
1035         .addComponent(snmptipo_cbox,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
1036         .addComponent(snmplbl_nombre))
1037
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
1038         .addComponent(snmplbl_titulo)
1039
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
1040         .addComponent(snmplbl_vistas)
1041
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
1042         .addComponent(snmptipo_cbox_vistas,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
1043
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
1044         .addComponent(snmplbl_listas)
1045
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
1046         .addComponent(snmptipo_cbox_listas,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
1047
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
1048         .addComponent(snmptitulo_scrpnl_info,
javax.swing.GroupLayout.PREFERRED_SIZE, 304,
javax.swing.GroupLayout.PREFERRED_SIZE)
1049
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
1050
.addGroup(snmplbl_comunidadesLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
1051         .addComponent(snmptitulo_aniadir)
1052         .addComponent(snmptitulo_eliminar))
1053         .addGap(54, 54, 54)
1054     );
1055
1056     tab_principal.addTab("SNMP", snmplbl_comunidades);
1057
1058     sshlblRSA.setText("Llave RSA");
1059

```

```

1060     ssh_lbl_modulo.setText("Tamaño de módulo");
1061
1062     ssh_spn_tamano.setModel(new javax.swing.SpinnerNumberModel(360,
360, 2048, 1));
1063
1064     ssh_txta_llaves.setColumns(20);
1065     ssh_txta_llaves.setEditable(false);
1066     ssh_txta_llaves.setRows(5);
1067     ssh_scrpnl_llaves.setViewportViewView(ssh_txta_llaves);
1068
1069     ssh_lbl_anfitrión.setText("Anfitrión");
1070
1071     ssh_lbl_dominio.setText("Dominio");
1072
1073     ssh_btn_aniadir.setText("Añadir");
1074     ssh_btn_aniadir.addActionListener(new java.awt.event.ActionListener() {
1075         public void actionPerformed(java.awt.event.ActionEvent evt) {
1076             ssh_btn_aniadirActionPerformed(evt);
1077         }
1078     });
1079
1080     ssh_check_nuevaLlave.setText("Generar nueva llave.");
1081
1082     jButton2.setText("Mostar llave actual");
1083     jButton2.addActionListener(new java.awt.event.ActionListener() {
1084         public void actionPerformed(java.awt.event.ActionEvent evt) {
1085             jButton2ActionPerformed(evt);
1086         }
1087     });
1088
1089     javax.swing.GroupLayout pan_sshLayout = new
javax.swing.GroupLayout(pan_ssh);
1090     pan_ssh.setLayout(pan_sshLayout);
1091     pan_sshLayout.setHorizontalGroup(
1092
pan_sshLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
1093         .addGroup(pan_sshLayout.createSequentialGroup()
1094             .addContainerGap()
1095
.addGroup(pan_sshLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.L
EADING)
1096             .addComponent(ssh_scrpnl_llaves,
javax.swing.GroupLayout.PREFERRED_SIZE, 431,
javax.swing.GroupLayout.PREFERRED_SIZE)
1097             .addComponent(ssh_lbl_RSA)
1098             .addComponent(jButton2)
1099             .addComponent(ssh_lbl_anfitrión)
1100             .addComponent(ssh_lbl_dominio)

```

```

1101 .addGroup(pan_sshLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.T
RAILING, false)
1102         .addComponent(ssh_txtf_dominio,
javax.swing.GroupLayout.Alignment.LEADING)
1103         .addComponent(ssh_txtf_anfitrión,
javax.swing.GroupLayout.Alignment.LEADING,
javax.swing.GroupLayout.PREFERRED_SIZE, 152,
javax.swing.GroupLayout.PREFERRED_SIZE))
1104         .addComponent(ssh_btn_aniadir)
1105         .addGroup(pan_sshLayout.createSequentialGroup())
1106         .addComponent(ssh_lbl_modulo)
1107
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
1108         .addComponent(ssh_spn_tamaño,
javax.swing.GroupLayout.PREFERRED_SIZE, 74,
javax.swing.GroupLayout.PREFERRED_SIZE))
1109         .addComponent(ssh_check_nuevaLlave)
1110         .addContainerGap(181, Short.MAX_VALUE))
1111 );
1112 pan_sshLayout.setVerticalGroup(
1113
pan_sshLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
1114     .addGroup(pan_sshLayout.createSequentialGroup())
1115     .addContainerGap()
1116     .addComponent(ssh_lbl_RSA)
1117     .addGap(33, 33, 33)
1118     .addComponent(jButton2)
1119
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
1120     .addComponent(ssh_scrpnl_llaves,
javax.swing.GroupLayout.DEFAULT_SIZE, 265, Short.MAX_VALUE)
1121
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
1122     .addComponent(ssh_check_nuevaLlave)
1123
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
1124
.addGroup(pan_sshLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.B
ASELINE)
1125     .addComponent(ssh_lbl_modulo)
1126     .addComponent(ssh_spn_tamaño,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
1127
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
1128     .addComponent(ssh_lbl_anfitrión)
1129
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

```

```

1130         .addComponent(ssh_txtf_anfitrion,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
1131
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
1132         .addComponent(ssh_lbl_dominio)
1133
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
1134         .addComponent(ssh_txtf_dominio,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
1135
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
1136         .addComponent(ssh_btn_aniadir)
1137         .addContainerGap()
1138     );
1139
1140     tab_principal.addTab("SSH", pan_ssh);
1141
1142     ntp_lbl_tipoServicio.setText("Tipo de servicio");
1143
1144     ntp_cbox_tipoServicio.setModel(new
javax.swing.DefaultComboBoxModel(new String[] { "NTP", "SNTP" }));
1145     ntp_cbox_tipoServicio.setEnabled(false);
1146     ntp_cbox_tipoServicio.addItemListener(new java.awt.event.ItemListener() {
1147         public void itemStateChanged(java.awt.event.ItemEvent evt) {
1148             ntp_cbox_tipoServicioItemStateChanged(evt);
1149         }
1150     });
1151
1152     ntp_lbl_nombreServicio.setText("Nombre o ip del servidor ");
1153
1154     ntp_txtf_nombreServicio.addActionListener(new
java.awt.event.ActionListener() {
1155         public void actionPerformed(java.awt.event.ActionEvent evt) {
1156             ntp_txtf_nombreServicioActionPerformed(evt);
1157         }
1158     });
1159
1160     ntp_check_prefered.setText("Preferida");
1161
1162     ntp_check_key.setText("Autenticación con llave");
1163     ntp_check_key.addItemListener(new java.awt.event.ItemListener() {
1164         public void itemStateChanged(java.awt.event.ItemEvent evt) {
1165             ntp_check_keyItemStateChanged(evt);
1166         }
1167     });
1168

```

```

1169     ntp_txtf_numeroLlave.setEnabled(false);
1170
1171     ntp_lbl_numeroLlave.setText("Número de llave");
1172
1173     ntp_lbl_valorLlave.setText("Valor de la llave");
1174
1175     ntp_txtf_valorLlave.setEnabled(false);
1176
1177     ntp_btn_aniadir.setText("Añadir");
1178     ntp_btn_aniadir.addActionListener(new java.awt.event.ActionListener() {
1179         public void actionPerformed(java.awt.event.ActionEvent evt) {
1180             ntp_btn_aniadirActionPerformed(evt);
1181         }
1182     });
1183
1184     ntp_tbl_servidores.setModel(new javax.swing.table.DefaultTableModel(
1185         new Object [][] {
1186             {null, null, null, null},
1187             {null, null, null, null},
1188             {null, null, null, null},
1189             {null, null, null, null}
1190         },
1191         new String [] {
1192             "Title 1", "Title 2", "Title 3", "Title 4"
1193         }
1194     ));
1195     ntp_scrpnl_servidores.setViewportViewView(ntp_tbl_servidores);
1196
1197     ntp_btn_eliminar.setText("Eliminar");
1198     ntp_btn_eliminar.addActionListener(new java.awt.event.ActionListener() {
1199         public void actionPerformed(java.awt.event.ActionEvent evt) {
1200             ntp_btn_eliminarActionPerformed(evt);
1201         }
1202     });
1203
1204     javax.swing.GroupLayout pan_ntpLayout = new
javax.swing.GroupLayout(pan_ntp);
1205     pan_ntp.setLayout(pan_ntpLayout);
1206     pan_ntpLayout.setHorizontalGroup(
1207     pan_ntpLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
1208         .addGroup(pan_ntpLayout.createSequentialGroup()
1209             .addGroup(pan_ntpLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
1210                 .addGroup(pan_ntpLayout.createSequentialGroup()
1211                     .addGroup(pan_ntpLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
1212                         .addComponent(ntp_check_key)
1213                         .addGap(471, Short.MAX_VALUE))
1214                     .addGroup(pan_ntpLayout.createSequentialGroup()

```

```

1215         .addComponent(ntp_lbl_numeroLlave)
1216         .addContainerGap(535, Short.MAX_VALUE))
1217         .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
pan_ntpLayout.createSequentialGroup())
1218
.addGroup(pan_ntpLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.T
RAILING)
1219         .addComponent(ntp_txf_valorLlave,
javax.swing.GroupLayout.Alignment.LEADING,
javax.swing.GroupLayout.DEFAULT_SIZE, 143, Short.MAX_VALUE)
1220         .addComponent(ntp_txf_numeroLlave,
javax.swing.GroupLayout.Alignment.LEADING,
javax.swing.GroupLayout.DEFAULT_SIZE, 143, Short.MAX_VALUE)
1221         .addGroup(javax.swing.GroupLayout.Alignment.LEADING,
pan_ntpLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING,
false)
1222         .addComponent(ntp_cbox_tipoServicio, 0, 143,
Short.MAX_VALUE)
1223         .addComponent(ntp_lbl_tipoServicio)
1224         .addComponent(ntp_lbl_nombreServicio)
1225         .addComponent(ntp_txf_nombreServicio)))
1226
.addGroup(pan_ntpLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.L
EADING)
1227         .addGroup(pan_ntpLayout.createSequentialGroup())
1228         .addGap(34, 34, 34)
1229         .addComponent(ntp_check_prefered))
1230         .addGroup(pan_ntpLayout.createSequentialGroup())
1231         .addGap(56, 56, 56)
1232         .addComponent(ntp_btn_aniadir)))
1233         .addGap(590, 590, 590))
1234         .addGroup(pan_ntpLayout.createSequentialGroup())
1235         .addComponent(ntp_btn_eliminar)
1236         .addContainerGap(543, Short.MAX_VALUE))
1237         .addGroup(pan_ntpLayout.createSequentialGroup())
1238         .addComponent(ntp_scrpnl_servidores,
javax.swing.GroupLayout.PREFERRED_SIZE, 581,
javax.swing.GroupLayout.PREFERRED_SIZE)
1239         .addContainerGap())
1240         .addGroup(pan_ntpLayout.createSequentialGroup())
1241         .addComponent(ntp_lbl_valorLlave)
1242         .addContainerGap(537, Short.MAX_VALUE))))
1243     );
1244     pan_ntpLayout.setVerticalGroup(
1245
pan_ntpLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
1246         .addGroup(pan_ntpLayout.createSequentialGroup())
1247         .addContainerGap()
1248         .addComponent(ntp_lbl_tipoServicio)

```

```
1249
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
1250     .addComponent(ntp_cbox_tipoServicio,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
1251
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
1252     .addComponent(ntp_lbl_nombreServicio)
1253
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
1254
.addGroup(pan_ntpLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.B
ASELINE)
1255     .addComponent(ntp_txf_nombreServicio,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
1256     .addComponent(ntp_check_prefered))
1257     .addGap(62, 62, 62)
1258     .addComponent(ntp_check_key)
1259
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
1260     .addComponent(ntp_lbl_numeroLlave)
1261
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
1262     .addComponent(ntp_txf_numeroLlave,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
1263
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
1264     .addComponent(ntp_lbl_valorLlave)
1265
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
1266
.addGroup(pan_ntpLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.B
ASELINE)
1267     .addComponent(ntp_txf_valorLlave,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
1268     .addComponent(ntp_btn_aniadir))
1269
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
1270     .addComponent(ntp_scrpnl_servidores,
javax.swing.GroupLayout.PREFERRED_SIZE, 130,
javax.swing.GroupLayout.PREFERRED_SIZE)
1271
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
```

```

1272         .addComponent(ntp_btn_eliminar)
1273         .addContainerGap(96, Short.MAX_VALUE))
1274     );
1275
1276     tab_principal.addTab("NTP", pan_ntp);
1277
1278     vis_lbl_nombre.setText("Nombre de la vista");
1279
1280     vis_btn_eliminar.setText("Eliminar");
1281     vis_btn_eliminar.addActionListener(new java.awt.event.ActionListener() {
1282         public void actionPerformed(java.awt.event.ActionEvent evt) {
1283             vis_btn_eliminarActionPerformed(evt);
1284         }
1285     });
1286
1287     vis_btn_aniadir.setText("Añadir");
1288     vis_btn_aniadir.addActionListener(new java.awt.event.ActionListener() {
1289         public void actionPerformed(java.awt.event.ActionEvent evt) {
1290             vis_btn_aniadirActionPerformed(evt);
1291         }
1292     });
1293
1294     vis_tbl_info.setModel(new javax.swing.table.DefaultTableModel(
1295         new Object [][] {
1296             {null, null, null, null},
1297             {null, null, null, null},
1298             {null, null, null, null},
1299             {null, null, null, null}
1300         },
1301         new String [] {
1302             "Title 1", "Title 2", "Title 3", "Title 4"
1303         }
1304     ));
1305     jScrollPane10.setViewportView(vis_tbl_info);
1306
1307     vis_lbl_vistas.setText("Vistas");
1308
1309     vis_lbl_mib.setText("MIB a incluir");
1310
1311     vis_btngrp_includedExcluded.add(vis_btn_included);
1312     vis_btn_included.setSelected(true);
1313     vis_btn_included.setText("Included");
1314
1315     vis_btngrp_includedExcluded.add(vis_btn_excluded);
1316     vis_btn_excluded.setText("Excluded");
1317
1318     javax.swing.GroupLayout pan_vistasLayout = new
javax.swing.GroupLayout(pan_vistas);
1319     pan_vistas.setLayout(pan_vistasLayout);
1320     pan_vistasLayout.setHorizontalGroup(

```

```

1321
pan_vistasLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING
)
1322     .addGroup(pan_vistasLayout.createSequentialGroup()
1323     .addContainerGap()
1324
.addGroup(pan_vistasLayout.createParallelGroup(javax.swing.GroupLayout.Alignment
.LEADING)
1325     .addComponent(jScrollPane10,
javax.swing.GroupLayout.DEFAULT_SIZE, 602, Short.MAX_VALUE)
1326     .addComponent(vis_lbl_vistas)
1327     .addGroup(pan_vistasLayout.createSequentialGroup()
1328     .addComponent(vis_lbl_nombre)
1329
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
1330     .addComponent(vis_txtf_nombre,
javax.swing.GroupLayout.PREFERRED_SIZE, 81,
javax.swing.GroupLayout.PREFERRED_SIZE)
1331
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
1332     .addComponent(vis_lbl_mib)
1333
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
1334     .addComponent(vis_txtf_MIB,
javax.swing.GroupLayout.PREFERRED_SIZE, 90,
javax.swing.GroupLayout.PREFERRED_SIZE)
1335
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
1336     .addComponent(vis_btn_included)
1337
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
1338     .addComponent(vis_btn_excluded))
1339     .addGroup(pan_vistasLayout.createSequentialGroup()
1340     .addComponent(vis_btn_aniadir)
1341
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
1342     .addComponent(vis_btn_eliminar)))
1343     .addContainerGap()
1344 );
1345 pan_vistasLayout.setVerticalGroup(
1346
pan_vistasLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING
)
1347     .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
pan_vistasLayout.createSequentialGroup()
1348     .addContainerGap()
1349     .addComponent(vis_lbl_vistas)
1350
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

```

```

1351
.addGroup(pan_vistasLayout.createParallelGroup(javax.swing.GroupLayout.Alignment
.BASELINE)
1352         .addComponent(vis_lbl_nombre)
1353         .addComponent(vis_txtf_nombre,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
1354         .addComponent(vis_lbl_mib)
1355         .addComponent(vis_txtf_MIB,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
1356         .addComponent(vis_btn_included)
1357         .addComponent(vis_btn_excluded))
1358
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
1359         .addComponent(jScrollPane10,
javax.swing.GroupLayout.DEFAULT_SIZE, 439, Short.MAX_VALUE)
1360
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
1361
.addGroup(pan_vistasLayout.createParallelGroup(javax.swing.GroupLayout.Alignment
.BASELINE)
1362         .addComponent(vis_btn_aniadir)
1363         .addComponent(vis_btn_eliminar))
1364         .addContainerGap()
1365     );
1366
1367     tab_principal.addTab("Vistas", pan_vistas);
1368
1369     acl_tbl_info.setModel(new javax.swing.table.DefaultTableModel(
1370         new Object [][] {
1371             {null, null, null, null},
1372             {null, null, null, null},
1373             {null, null, null, null},
1374             {null, null, null, null}
1375         },
1376         new String [] {
1377             "Title 1", "Title 2", "Title 3", "Title 4"
1378         }
1379     ));
1380     acl_scrpnl_info.setViewportView(acl_tbl_info);
1381
1382     acl_lbl_host.setText("Host a integrar en la lista");
1383
1384     acl_spn_numerolista.setModel(new javax.swing.SpinnerNumberModel(1, 1,
200, 1));
1385
1386     acl_lbl_numerolista.setText("Número de la lista");

```



```

1428
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
1429         .addComponent(ac1_spn_numerolista,
javax.swing.GroupLayout.PREFERRED_SIZE, 74,
javax.swing.GroupLayout.PREFERRED_SIZE)))
1430
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
1431
.addGroup(pan_ac1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.L
EADING)
1432         .addComponent(ac1_btn_permit)
1433         .addGroup(pan_ac1Layout.createSequentialGroup())
1434         .addComponent(ac1_btn_deny)
1435         .addGap(18, 18, 18)
1436         .addComponent(ac1_lbl_host)
1437         .addGap(19, 19, 19)
1438         .addComponent(ac1_txf_host,
javax.swing.GroupLayout.PREFERRED_SIZE, 90,
javax.swing.GroupLayout.PREFERRED_SIZE)))
1439
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, 149,
Short.MAX_VALUE)))
1440         .addContainerGap()
1441     );
1442     pan_ac1Layout.setVerticalGroup(
1443
pan_ac1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
1444         .addGroup(pan_ac1Layout.createSequentialGroup())
1445         .addContainerGap()
1446
.addGroup(pan_ac1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.T
RAILING)
1447         .addGroup(pan_ac1Layout.createSequentialGroup())
1448         .addComponent(ac1_btn_permit)
1449
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
1450         .addComponent(ac1_btn_deny))
1451         .addGroup(pan_ac1Layout.createSequentialGroup())
1452         .addComponent(ac1_lbl_acl)
1453
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
1454
.addGroup(pan_ac1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.B
ASELINE)
1455         .addComponent(ac1_lbl_numerolista)
1456         .addComponent(ac1_spn_numerolista,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)

```

```

1457         .addComponent(acl_txtf_host,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
1458         .addComponent(acl_lbl_host))))
1459
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
1460         .addComponent(acl_scrpnl_info,
javax.swing.GroupLayout.DEFAULT_SIZE, 437, Short.MAX_VALUE)
1461
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
1462
.addGroup(pan_aclLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.B
ASELINE)
1463         .addComponent(acl_btn_aniadir)
1464         .addComponent(acl_btn_eliminar))
1465     .addContainerGap()
1466 );
1467
1468     tab_principal.addTab("ACL", pan_acl);
1469
1470     aaut_btn_comenzar.setText("Comenzar");
1471     aaut_btn_comenzar.addActionListener(new java.awt.event.ActionListener()
{
1472         public void actionPerformed(java.awt.event.ActionEvent evt) {
1473             aaut_btn_comenzarActionPerformed(evt);
1474         }
1475     });
1476
1477     aauto_txta_info.setColumns(20);
1478     aauto_txta_info.setRows(5);
1479     aaut_scrpnl_info.setViewportView(aauto_txta_info);
1480
1481     aaut_btn_enviar.setText("Enviar");
1482     aaut_btn_enviar.addActionListener(new java.awt.event.ActionListener() {
1483         public void actionPerformed(java.awt.event.ActionEvent evt) {
1484             aaut_btn_enviarActionPerformed(evt);
1485         }
1486     });
1487
1488     jButton3.setText("Cancelar");
1489     jButton3.addActionListener(new java.awt.event.ActionListener() {
1490         public void actionPerformed(java.awt.event.ActionEvent evt) {
1491             jButton3ActionPerformed(evt);
1492         }
1493     });
1494
1495     javax.swing.GroupLayout pan_auto2Layout = new
javax.swing.GroupLayout(pan_auto2);
1496     pan_auto2.setLayout(pan_auto2Layout);

```

```

1497     pan_auto2Layout.setHorizontalGroup(
1498
pan_auto2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING
)
1499         .addGroup(pan_auto2Layout.createSequentialGroup())
1500         .addContainerGap()
1501
.addGroup(pan_auto2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.
LEADING)
1502         .addGroup(pan_auto2Layout.createSequentialGroup())
1503         .addComponent(aaut_btn_comenzar)
1504
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, 304,
Short.MAX_VALUE)
1505         .addComponent(jButton3))
1506         .addComponent(aaut_scrpnl_info,
javax.swing.GroupLayout.PREFERRED_SIZE, 459,
javax.swing.GroupLayout.PREFERRED_SIZE))
1507
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
1508
.addGroup(pan_auto2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.
TRAILING)
1509         .addComponent(aaut_btn_enviar,
javax.swing.GroupLayout.DEFAULT_SIZE, 136, Short.MAX_VALUE)
1510         .addComponent(aaut_txtf_message,
javax.swing.GroupLayout.DEFAULT_SIZE, 136, Short.MAX_VALUE))
1511         .addContainerGap()
1512     );
1513     pan_auto2Layout.setVerticalGroup(
1514
pan_auto2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING
)
1515         .addGroup(pan_auto2Layout.createSequentialGroup())
1516         .addContainerGap()
1517
.addGroup(pan_auto2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.
BASELINE)
1518         .addComponent(aaut_btn_comenzar)
1519         .addComponent(jButton3))
1520
.addGroup(pan_auto2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.
LEADING)
1521         .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
pan_auto2Layout.createSequentialGroup())
1522
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
1523         .addComponent(aaut_txtf_message,
javax.swing.GroupLayout.PREFERRED_SIZE,

```

```

javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
1524
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
1525     .addComponent(aaut_btn_enviar))
1526     .addGroup(pan_auto2Layout.createSequentialGroup())
1527     .addGap(7, 7, 7)
1528     .addComponent(aaut_scrpnl_info,
javax.swing.GroupLayout.DEFAULT_SIZE, 484, Short.MAX_VALUE)))
1529     .addContainerGap()
1530 );
1531
1532 tab_principal.addTab("Auto asegurar", pan_auto2);
1533
1534 lbl_tareas.setText("Tareas a realizar");
1535
1536 btn_ejecutar.setText("Ejecutar");
1537 btn_ejecutar.addActionListener(new java.awt.event.ActionListener() {
1538     public void actionPerformed(java.awt.event.ActionEvent evt) {
1539         btn_ejecutarActionPerformed(evt);
1540     }
1541 });
1542
1543 btn_desconectar.setText("Desconectar");
1544 btn_desconectar.addActionListener(new java.awt.event.ActionListener() {
1545     public void actionPerformed(java.awt.event.ActionEvent evt) {
1546         btn_desconectarActionPerformed(evt);
1547     }
1548 });
1549
1550 jScrollPane2.setViewportView(lst_tareas);
1551
1552 javax.swing.GroupLayout layout = new
javax.swing.GroupLayout(getContentPane());
1553 getContentPane().setLayout(layout);
1554 layout.setHorizontalGroup(
1555
layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
1556     .addGroup(layout.createSequentialGroup())
1557
.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING
)
1558     .addGroup(layout.createSequentialGroup())
1559     .addGap(14, 14, 14)
1560
.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING
)
1561     .addComponent(lbl_tareas)

```

```

1562         .addComponent(tab_principal,
javax.swing.GroupLayout.PREFERRED_SIZE, 627,
javax.swing.GroupLayout.PREFERRED_SIZE)))
1563         .addGroup(layout.createSequentialGroup()
1564         .addContainerGap()
1565         .addComponent(btn_ejecutar)
1566         .addGap(60, 60, 60)
1567         .addComponent(pgrb_ejecutar,
javax.swing.GroupLayout.PREFERRED_SIZE, 210,
javax.swing.GroupLayout.PREFERRED_SIZE)
1568         .addGap(44, 44, 44)
1569         .addComponent(btn_desconectar))
1570         .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
layout.createSequentialGroup()
1571         .addGap(14, 14, 14)
1572         .addComponent(jScrollPane2,
javax.swing.GroupLayout.DEFAULT_SIZE, 627, Short.MAX_VALUE)))
1573         .addContainerGap()
1574     );
1575     layout.setVerticalGroup(
1576     layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
1577         .addGroup(layout.createSequentialGroup()
1578         .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE)
1579         .addComponent(tab_principal,
javax.swing.GroupLayout.PREFERRED_SIZE, 580,
javax.swing.GroupLayout.PREFERRED_SIZE)
1580         .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
1581         .addComponent(lbl_tareas)
1582         .addGap(1, 1, 1)
1583         .addComponent(jScrollPane2,
javax.swing.GroupLayout.PREFERRED_SIZE, 69,
javax.swing.GroupLayout.PREFERRED_SIZE)
1584         .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
1585         .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)
1586         .addComponent(pgrb_ejecutar,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
1587         .addComponent(btn_ejecutar)
1588         .addComponent(btn_desconectar))
1589         .addContainerGap()
1590     );
1591
1592     pack();

```

```

1593 }// </editor-fold>
1594
1595 private void btn_ejecutarActionPerformed(java.awt.event.ActionEvent evt) {
1596     //pgrb_ejecutar.setIndeterminate(true);
1597
1598     // Task reconectar = new Task("reconectar");
1599     // Task desconectar = new Task("desconectar");
1600
1601     //listaDeTareas.add(reconectar);
1602     //listaDeTareas.add(desconectar);
1603
1604     //listaModelo.add(0, "Reconectar");
1605     //listaModelo.addElement("Desconectar");
1606     //lst_tareas.setModel(listaModelo);
1607
1608     pgrb_ejecutar.setMaximum(listaDeTareas.size());
1609
1610
1611
1612     Iterator<Task> iterador = listaDeTareas.iterator();
1613     boolean todasOperacionesOK = true;
1614     try {
1615         for (int i = 0; i < listaDeTareas.size(); i++) {
1616
1617             Task temp = (Task) listaDeTareas.get(i);
1618             if (temp.exe()) {
1619                 listaModelo.remove(listaModeloExe);
1620                 listaModelo.add(listaModeloExe, temp.getTask() + "\tOK");
1621                 lst_tareas.setModel(listaModelo);
1622                 listaModeloExe++;
1623
1624                 pgrb_ejecutar.setValue(i + 1);
1625             } else {
1626                 todasOperacionesOK = false;
1627                 listaModelo.remove(i);
1628                 listaModelo.add(i, temp.getTask() + "\tX");
1629                 lst_tareas.setModel(listaModelo);
1630                 JOptionPane.showMessageDialog(pan_lineas, "Error en tarea" +
temp.getTask());
1631             }
1632         }
1633         pgrb_ejecutar.setValue(i + 1);
1634         lst_tareas.ensureIndexIsVisible(lst_tareas.getModel().getSize() - 1);
1635     }
1636     lst_tareas.setModel(listaModelo);
1637     if (todasOperacionesOK) {
1638         JOptionPane.showMessageDialog(pan_lineas, "Todas las operaciones se
realizaron con exito");
1639     }
1640     lst_tareas.ensureIndexIsVisible(lst_tareas.getModel().getSize() - 1);

```

```

1641     listaDeTareas = new ArrayList<Task>();
1642     pgrb_ejecutar.setValue(0);
1643
1644     load();
1645
1646     this.repaint();
1647     //listaModelo.clear();
1648
1649
1650 } catch (Exception e) {
1651     JOptionPane.showMessageDialog(pan_lineas, e);
1652 }
1653
1654 // pgrb_ejecutar.setIndeterminate(false);
1655 }
1656
1657 private void btn_desconectarActionPerformed(java.awt.event.ActionEvent evt)
1658 {
1659     try {
1660         Task desconectar = new Task("Desconectar", null);
1661         desconectar.exe();
1662         MainGUI.this.setVisible(false);
1663         MainGUI.this.dispose();
1664     } catch (Exception e) {
1665         JOptionPane.showMessageDialog(pan_lineas, e);
1666     }
1667 }
1668 private void PRI_chkbtn_resetActionPerformed(java.awt.event.ActionEvent
1669 evt) {
1670     PRI_spn_nivel.setEnabled(false);
1671 }
1672 private void PRI_chkbtn_allActionPerformed(java.awt.event.ActionEvent evt)
1673 {
1674     PRI_spn_nivel.setEnabled(true);
1675 }
1676 private void PRI_btn_aniadirActionPerformed(java.awt.event.ActionEvent evt)
1677 {
1678     String[] privilegeParameters = new String[3];
1679     if (PRI_chkbtn_all.isSelected()) {
1680         privilegeParameters[0] = "all";
1681     }
1682     if (PRI_chkbtn_reset.isSelected()) {
1683         privilegeParameters[0] = "reset";
1684     } else {
1685         privilegeParameters[1] = PRI_spn_nivel.getValue().toString();
1686     }
1687     String comando = PRI_txt_comando.getText();

```

```

1687     if (comando.equals("")) {
1688         JOptionPane.showMessageDialog(pan_conexion, "Error no comando");
1689     } else {
1690         privilegeParameters[2] = comando;
1691     }
1692     Task newPrivilege = new Task("newPrivilege", privilegeParameters);
1693     actualizaListaTareas(newPrivilege);
1694 }
1695
1696 private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
1697     if (PRI_lst_opciones.getSelectedIndex() != -1) {
1698         PRI_txt_comando.setText(PRI_txt_comando.getText() + " " +
1699 PRI_lst_opciones.getSelectedValue().toString());
1700     } else {
1701         JOptionPane.showMessageDialog(pan_conexion, "No hay nada
1702 seleccionado");
1703     }
1704 }
1705
1706 private void PRI_btn_opcionesActionPerformed(java.awt.event.ActionEvent
1707 evt) {
1708     String[] loadOptionsParameters = {
1709         PRI_txt_comando.getText()
1710     };
1711     try {
1712         Task reconectar = new Task("reconectar");
1713         Task desconectar = new Task("desconectar");
1714         Task loadOptions = new Task("loadOptions", loadOptionsParameters);
1715         reconectar.exe();
1716         loadOptions.exe();
1717         desconectar.exe();
1718     } catch (Exception e) {
1719         JOptionPane.showMessageDialog(pan_lineas, e);
1720     }
1721     String[] temp = data.getOptions();
1722     DefaultListModel listaModeloOpciones = new DefaultListModel();
1723     int intTempLength = temp.length;
1724     for (int i = 0; i < intTempLength; i++) {
1725         listaModeloOpciones.addElement(temp[i]);
1726     }
1727     PRI_lst_opciones.setModel(listaModeloOpciones);
1728 }
1729
1730 private void NIV_btn_aniadirActionPerformed(java.awt.event.ActionEvent evt)
1731 {
1732     int rowIndex = NIV_tbl_niveles.getSelectedRow();
1733     if (rowIndex == -1) {

```

```

1732     JOptionPane.showMessageDialog(pan_lineas, "Error: No hay elementos
seleccionados.");
1733     } else {
1734
1735     if (!NIV_txt_nContrasenia.getText().equals("")) {
1736         //System.out.println();
1737         String strLevel[] = {
1738             NIV_tbl_niveles.getValueAt(rowIndex, 0).toString(),
1739             NIV_txt_nContrasenia.getText()
1740         };
1741         Task levelPassword = new Task("levelPassword", strLevel);
1742         actualizaListaTareas(levelPassword);
1743     } else {
1744         JOptionPane.showMessageDialog(pan_lineas, "Error: Falta
contraseña.");
1745     }
1746     } // TODO add your handling code here:
1747 }
1748
1749 private void ACC_txt_numerosActionPerformed(java.awt.event.ActionEvent
evt) {
1750     // TODO add your handling code here:
1751 }
1752
1753 private void ACC_btn_aniadirActionPerformed(java.awt.event.ActionEvent
evt) {
1754     String[] newLogTriesParameters = {
1755         ACC_spn_intentos.getValueAt().toString(),};
1756
1757     Task numberLogTries = new Task("numberLogTries",
newLogTriesParameters);
1758     actualizaListaTareas(numberLogTries);
1759 }
1760
1761 private void CON_bytn_aniadirActionPerformed(java.awt.event.ActionEvent
evt) {
1762     String[] conexionData = {
1763         CON_cbox_lineas.getSelectedItem().toString(),
1764         CON_spin_min.getValueAt().toString(),
1765         CON_spin_sec.getValueAt().toString()
1766     };
1767     Task newConexionTime = new Task("newConnexionTime", conexionData);
1768     actualizaListaTareas(newConexionTime);
1769 }
1770
1771 private void US_btn_aniadirActionPerformed(java.awt.event.ActionEvent evt)
{
1772     String[] userData = {
1773         US_txt_usuario.getText(),
1774         US_sld_privilegios.getValueAt() + "",

```

```

1775     US_cbox_encriptacion.getSelectedItemAt().toString(),
1776     US_txt_contrasenia.getText()
1777     };
1778     Task newUser = new Task("newUser", userData);
1779     actualizaListaTareas(newUser);
1780 }
1781
1782 private void US_sld_privilegiosStateChanged(javax.swing.event.ChangeEvent
1783 evt) {
1784     US_lbl_noPrivilegio.setText("" + US_sld_privilegios.getValue());
1785 }
1786 private void US_btn_eliminarActionPerformed(java.awt.event.ActionEvent evt)
1787 {
1788     int rowIndex = US_tbl_usuarios.getSelectedRow();
1789     if (rowIndex == -1) {
1790         JOptionPane.showMessageDialog(pan_lineas, "Error: No hay elementos
1791 seleccionados.");
1792     } else {
1793         // System.out.println();
1794         String strUser[] = {US_tbl_usuarios.getValueAt(rowIndex, 0).toString()};
1795         Task deleteUser = new Task("deleteUser", strUser);
1796         actualizaListaTareas(deleteUser);
1797     }
1798 }
1799 private void LIN_btn_aniadirActionPerformed(java.awt.event.ActionEvent evt)
1800 {
1801     try {
1802         if (LIN_txt_nContrasenia.getText().equals("")) {
1803             JOptionPane.showMessageDialog(pan_lineas, "Error no contraseña.");
1804         } else {
1805             String strLineas[] = {
1806                 (String) LIN_cbox_linea.getSelectedItem(),
1807                 LIN_txt_nContrasenia.getText()
1808             };
1809             Task setPassLin = new Task("setPasswordLin", strLineas);
1810             actualizaListaTareas(setPassLin);
1811         }
1812     } catch (Exception e) {
1813         JOptionPane.showMessageDialog(pan_lineas, e);
1814     }
1815 }
1816 }
1817
1818 private void
1819 SCR_rbtn_deshabilitarActionPerformed(java.awt.event.ActionEvent evt) {
1820     //SCR_txt_SecretPassword.setEnabled(false);

```

```

1820     SCR_txt_SecretPassword.setEnabled(false);
1821     if (showRunInfo.secretHabilitado()) {
1822         //SCR_lbl_solicitud.setText("Proporciona contraseña secret.");
1823     } else {
1824         SCR_lbl_solicitud.setText("-");
1825         JOptionPane.showMessageDialog(pan_lineas, "No hay otra forma de
habilitar nivel administrador.");
1826
1827         SCR_txt_SecretPassword.setText("");
1828     }
1829 }
1830
1831 private void SCR_rbtn_habilitarActionPerformed(java.awt.event.ActionEvent
evt) {
1832     SCR_txt_SecretPassword.setEnabled(true);
1833     SCR_lbl_solicitud.setText("Proporciona nueva contraseña.");
1834 }
1835
1836 private void SCT_btn_aniadirActionPerformed(java.awt.event.ActionEvent evt)
{
1837     try {
1838         if (SCR_txt_SecretPassword.isEnabled() &&
SCR_txt_SecretPassword.getText().equals("")) {
1839             JOptionPane.showMessageDialog(pan_lineas, "Error no contraseña");
1840         } else {
1841             String strsecret[] = {SCR_txt_SecretPassword.getText()};
1842             if (SCR_rbtn_habilitar.isSelected()) {
1843                 Task enableSecret = new Task("enablePassword", strsecret);
1844                 actualizaListaTareas(enableSecret);
1845
1846             } else if (SCR_rbtn_deshabilitar.isSelected()) {
1847                 Task disableSecret = new Task("disablePassword");
1848                 actualizaListaTareas(disableSecret);
1849             } else {
1850                 JOptionPane.showMessageDialog(pan_lineas, "Error selecciona
algo");
1851             }
1852         }
1853     } catch (Exception e) {
1854         JOptionPane.showMessageDialog(pan_lineas, e);
1855     }
1856 }
1857
1858 private void
btn_ShowTechSupportActionPerformed(java.awt.event.ActionEvent evt) {
1859     pgrb_ejecutar.setIndeterminate(true);
1860     JOptionPane.showMessageDialog(pan_lineas, "Toma algunos segundos.");
1861     try {
1862         Task reconnect = new Task("reconectar");
1863         Task loadTechSupport = new Task("loadShowTechSupport");

```

```

1864     Task desconectar = new Task("desconectar");
1865
1866     if (reconnect.exe()) {
1867
1868         if (loadTechSupport.exe()) {
1869             IG_txta_showTechSupport.setText(data.getBuffer());
1870             data.setBuffer(null);
1871             desconectar.exe();
1872         }
1873     }
1874
1875     } catch (Exception e) {
1876         JOptionPane.showMessageDialog(pan_lineas, e);
1877     }
1878     pgrb_ejecutar.setIndeterminate(false);
1879 }
1880
1881 private void
ntp_txtf_nombreServicioActionPerformed(java.awt.event.ActionEvent evt) {
1882     // TODO add your handling code here:
1883 }
1884
1885 private void vis_btn_aniadirActionPerformed(java.awt.event.ActionEvent evt) {
1886     String nombreVista = vis_txtf_nombre.getText();
1887     String MIB = vis_txtf_MIB.getText();
1888     String restriccion = "";
1889     if (vis_btn_included.isSelected()) {
1890         restriccion = "included";
1891     } else {
1892         restriccion = "excluded";
1893     }
1894     if (nombreVista.equals("") || MIB.equals("")) {
1895         JOptionPane.showMessageDialog(pan_conexion,
1896             "Campos vacios");
1897     } else {
1898         String[] parametros = {nombreVista, MIB, restriccion};
1899         Task addView = new Task("addView", parametros);
1900         actualizaListaTareas(addView);
1901     }
1902 }
1903
1904 private void MNS_btn_terminalActionPerformed(java.awt.event.ActionEvent
evt) {
1905     String[] tamanioBuffer = {MNS_spn_terminal.getValue().toString()};
1906     Task loggingBuffered = new Task("loggingBuffered", tamanioBuffer);
1907     actualizaListaTareas(loggingBuffered);
1908 }
1909
1910 private void MNS_btn_prioridadActionPerformed(java.awt.event.ActionEvent
evt) {

```

```

1911     String nivelMensajes = MNS_cbox_prioridad.getSelectedItemAt().toString();
1912
1913     if (nivelMensajes.equals("default")) {
1914         JOptionPane.showMessageDialog(pan_conexion, "Selecciona un nivel de
mensajes.");
1915     } else {
1916         String[] messageLevel = { nivelMensajes };
1917         Task loggingMonitor = new Task("loggingMonitor", messageLevel);
1918         actualizaListaTareas(loggingMonitor);
1919     }
1920 }
1921 public void update(Observable obj, Object arg) {
1922     if (data.isReading()) {
1923         if (arg instanceof String) {
1924
1925             // System.out.println("Reading flag " + arg.toString());
1926             javax.swing.table.TableModel model = MNS_tbl_mensajes.getModel();
1927             System.out.println("mensaje " + arg.toString());
1928             int columnas = MNS_tbl_mensajes.getColumnCount();
1929             int filas = MNS_tbl_mensajes.getRowCount() + 1;
1930
1931
1932             String[][] tablaTemp = new String[filas][columnas];
1933
1934             String[] lineaTemp = arg.toString().split(" ", 5);
1935
1936             tablaTemp[0][0] = lineaTemp[0].substring(1, lineaTemp[0].length())
+ " " + lineaTemp[2];
1937             tablaTemp[0][1] = lineaTemp[3];
1938             tablaTemp[0][2] = lineaTemp[4];
1939
1940             for (int i = 1; i <= filas - 1; i++) {
1941                 for (int j = 0; j <= columnas - 1; j++) {
1942                     tablaTemp[i][j] = (String) MNS_tbl_mensajes.getValueAt(i - 1, j);
1943                 }
1944             }
1945
1946             String[] titulos = {
1947                 "Fecha", "Hora", "Evento"
1948             };
1949
1950             DefaultTableModel tablaModelo = new DefaultTableModel(tablaTemp,
titulos);
1951
1952             MNS_tbl_mensajes.setModel(tablaModelo);
1953             try {
1954             } catch (Exception e) {
1955                 System.out.println(e.getStackTrace());
1956             }
1957         } else {
1958             System.out.println("NameObserver: Some other change to subject!");

```

```

1959     }
1960   } else if (data.isReadingAll()) {
1961     String temp = aauto_txta_info.getText();
1962     temp = temp + data.getMessageBuffer();
1963     aauto_txta_info.setText(temp);
1964   }
1965
1966
1967   }
1968 private void MNS_btn_comenzarActionPerformed(java.awt.event.ActionEvent
1969 evt) {
1970     try {
1971       /* Task reconectar = new Task("reconectar");
1972       String[] parametros0 = {"terminal length 0"};
1973       Task terminalLength = new Task("send", parametros0);
1974       Task terminalMonitor = new Task("terminalMonitor");
1975
1976       reconectar.exe();
1977
1978       terminalLength.exe();
1979       terminalMonitor.exe();*/
1980
1981       data.setReading(true);
1982       new Task("Observer").start();
1983       data.addObserver(this);
1984
1985     } catch (Exception e) {
1986       JOptionPane.showMessageDialog(pan_conexion, e);
1987     }
1988   }
1989
1990 private void MNS_btn_terminarActionPerformed(java.awt.event.ActionEvent
1991 evt) {
1992     Task desconectar = new Task("desconectar");
1993     try {
1994       data.setReading(false);
1995       data.deleteObserver(this);
1996       desconectar.exe();
1997     } catch (Exception e) {
1998       JOptionPane.showMessageDialog(pan_conexion, e);
1999     }
2000
2001 private void ntp_cbox_tipoServicioItemStateChanged(java.awt.event.ItemEvent
2002 evt) {
2003     if
2004     (ntp_cbox_tipoServicio.getSelectedItemAt().toString().equalsIgnoreCase("sntp")) {
2005       ntp_check_key.setEnabled(false);
2006       ntp_txtf_numeroLlave.setText("");

```

```

2005     ntp_txf_valorLlave.setText("");
2006     ntp_txf_numeroLlave.setEnabled(false);
2007     ntp_txf_valorLlave.setEnabled(false);
2008
2009     } else {
2010         ntp_check_key.setEnabled(true);
2011     }
2012 }
2013
2014 private void ntp_check_keyItemStateChanged(java.awt.event.ItemEvent evt) {
2015     if (ntp_check_key.isSelected()) {
2016         ntp_txf_numeroLlave.setEnabled(true);
2017         ntp_txf_valorLlave.setEnabled(true);
2018     } else {
2019         ntp_txf_numeroLlave.setEnabled(false);
2020         ntp_txf_valorLlave.setEnabled(false);
2021     }
2022 }
2023
2024 private void ntp_btn_aniadirActionPerformed(java.awt.event.ActionEvent evt)
{
2025     String ntpServer = new String("");
2026     String[] ntpParametros = new String[2];
2027     if
(ntp_cbox_tipoServicio.getSelectedItem().toString().equalsIgnoreCase("NTP")) {
2028         ntpServer = ntp_txf_nombreServicio.getText();
2029         if (ntpServer.equals("")) {
2030             JOptionPane.showMessageDialog(pan_conexion, "Proporciona ip o
nombre de un servicio.");
2031         } else {
2032             Pattern ipPattern = Pattern.compile("[0-9]+.[0-9]+.[0-9]+.[0-9]+");
2033             Matcher ipMatcher = ipPattern.matcher(ntpServer);
2034
2035             Pattern dotComPattern = Pattern.compile("[a-z]+.[a-z]{3}");
2036             Matcher dotComMatcher = dotComPattern.matcher(ntpServer);
2037
2038             if (true) {
2039                 ntpParametros[0] = ntpServer;
2040             } else {
2041                 JOptionPane.showMessageDialog(pan_conexion, "Formato del
servicio no coincide.");
2042             }
2043         }
2044         if (ntp_check_prefered.isSelected()) {
2045             ntpParametros[1] = "prefer";
2046         } else {
2047             ntpParametros[1] = "";
2048         }
2049     } else {
2050         //Caso SNTP

```

```

2051     }
2052
2053     Task ntp = new Task("ntpServer", ntpParametros);
2054     actualizaListaTareas(ntp);
2055 }
2056
2057 private void ntp_btn_eliminarActionPerformed(java.awt.event.ActionEvent evt)
{
2058     int rowIndex = ntp_tbl_servidores.getSelectedRow();
2059     if (rowIndex == -1) {
2060         JOptionPane.showMessageDialog(pan_lineas, "Error: No hay elementos
seleccionados.");
2061     } else {
2062         // System.out.println();
2063         String strNTPServer[] = {ntp_tbl_servidores.getValueAt(rowIndex,
1).toString()};
2064         Task deleteUser = new Task("deleteNTPServer", strNTPServer);
2065         actualizaListaTareas(deleteUser);
2066     }
2067 }
2068
2069 private void ssh_btn_aniadirActionPerformed(java.awt.event.ActionEvent evt)
{
2070     String[] parametrosSSH = new String[3];
2071     String anfitrión = ssh_txtf_anfitrión.getText();
2072     String dominio = ssh_txtf_dominio.getText();
2073     boolean error = false;
2074     if (anfitrión.equals("")) {
2075         JOptionPane.showMessageDialog(pan_conexion, "Nombre de anfitrión
vacío.");
2076         error = true;
2077     } else {
2078         parametrosSSH[0] = anfitrión;
2079     }
2080
2081     if (dominio.equals("")) {
2082         JOptionPane.showMessageDialog(pan_conexion, "Nombre de dominio
vacío.");
2083         error = true;
2084     } else {
2085         parametrosSSH[1] = anfitrión;
2086     }
2087
2088     if (ssh_check_nuevaLlave.isSelected()) {
2089         parametrosSSH[2] = ssh_spn_tamaño.getValue().toString();
2090     }
2091     if (!error) {
2092         Task configureSSH = new Task("configureSSH", parametrosSSH);
2093         actualizaListaTareas(configureSSH);
2094     }

```

```

2095 }
2096
2097 private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {
2098     Task reconectar = new Task("reconectar");
2099     Task loadCryptoKey = new Task("loadCryptoKey");
2100     Task desconectar = new Task("desconectar");
2101     try {
2102         reconectar.exe();
2103         loadCryptoKey.exe();
2104         desconectar.exe();
2105     } catch (Exception e) {
2106         JOptionPane.showMessageDialog(pan_conexion, e.getMessage());
2107     }
2108     ssh_txta_llaves.setText(data.getBuffer());
2109
2110
2111 }
2112
2113 private void snmp_btn_aniadirActionPerformed(java.awt.event.ActionEvent
2114 evt) {
2115     String nombre = snmp_txtf_nombre.getText();
2116     String tipo = snmp_cbox_tipo.getSelectedItem().toString();
2117     String vista = snmp_cbox_vistas.getSelectedItem().toString();
2118     String lista = snmp_cbox_listas.getSelectedItem().toString();
2119     String[] parametros = new String[4];
2120     if (nombre.equals("")) {
2121         JOptionPane.showMessageDialog(pan_conexion, "Nombre vacio.");
2122     } else {
2123         parametros[0] = nombre;
2124         parametros[1] = tipo;
2125         parametros[2] = vista;
2126         parametros[3] = lista;
2127         Task addSNMPCCommunity = new Task("addSNMPCCommunity",
2128 parametros);
2129         actualizaListaTareas(addSNMPCCommunity);
2130     }
2131 }
2132 private void vis_btn_eliminarActionPerformed(java.awt.event.ActionEvent evt)
2133 {
2134     int rowIndex = vis_tbl_info.getSelectedRow();
2135     if (rowIndex == -1) {
2136         JOptionPane.showMessageDialog(pan_lineas, "Error: No hay elementos
2137 seleccionados.");
2138     } else {
2139         // System.out.println();
2140         String parametros[] = {
2141             vis_tbl_info.getValueAt(rowIndex, 0).toString(),

```

```

2141         vis_tbl_info.getValueAt(rowIndex, 1).toString(),
2142         vis_tbl_info.getValueAt(rowIndex, 2).toString()
2143     };
2144     Task deleteView = new Task("removeView", parametros);
2145     actualizaListaTareas(deleteView);
2146 }
2147 }
2148
2149 private void acl_btn_aniadirActionPerformed(java.awt.event.ActionEvent evt) {
2150     String numero = acl_spn_numerolista.getValue().toString();
2151     String host = acl_txtf_host.getText();
2152     String restriccion = "";
2153     String[] parametros = new String[3];
2154     if (acl_btn_permit.isSelected()) {
2155         restriccion = "permit";
2156     } else {
2157         restriccion = "deny";
2158     }
2159     if (host.equals("")) {
2160         JOptionPane.showMessageDialog(pan_conexion, "Anfitrión vacío.");
2161     } else {
2162         parametros[0] = numero;
2163         parametros[1] = host;
2164         parametros[2] = restriccion;
2165         Task addACL = new Task("addACL", parametros);
2166         actualizaListaTareas(addACL);
2167     }
2168 }
2169
2170 private void acl_btn_eliminarActionPerformed(java.awt.event.ActionEvent evt)
2171 {
2172     int rowIndex = acl_tbl_info.getSelectedRow();
2173     if (rowIndex == -1) {
2174         JOptionPane.showMessageDialog(pan_lineas, "Error: No hay elementos
2175 seleccionados.");
2176     } else {
2177         // System.out.println();
2178         String parametros[] = {
2179             acl_tbl_info.getValueAt(rowIndex, 0).toString(),
2180             acl_tbl_info.getValueAt(rowIndex, 1).toString(),
2181             acl_tbl_info.getValueAt(rowIndex, 2).toString()
2182         };
2183         Task deleteACL = new Task("removeACL", parametros);
2184         actualizaListaTareas(deleteACL);
2185     }
2186 }
2187
2188 private void snmp_btn_eliminarActionPerformed(java.awt.event.ActionEvent
2189 evt) {

```

```

2188     int rowIndex = snmp_tbl_info.getSelectedRow();
2189     if (rowIndex == -1) {
2190         JOptionPane.showMessageDialog(pan_lineas, "Error: No hay elementos
seleccionados.");
2191     } else {
2192         // System.out.println();
2193 String nombre = snmp_tbl_info.getValueAt(rowIndex, 0).toString();
2194 String tipo = snmp_tbl_info.getValueAt(rowIndex, 1).toString();
2195 String vista = snmp_tbl_info.getValueAt(rowIndex, 2).toString();
2196 String acl = snmp_tbl_info.getValueAt(rowIndex, 3).toString();
2197     String parametros[] = {
2198         nombre, tipo, vista, acl
2199     };
2200     Task removeSNMPCommunity = new Task("removeSNMPCommunity",
parametros);
2201     actualizaListaTareas(removeSNMPCommunity);
2202     }
2203 }
2204
2205 private void aaut_btn_comenzarActionPerformed(java.awt.event.ActionEvent
evt) {
2206     try {
2207         Task reconectar = new Task("reconectar");
2208         String[] parametros0 = {"terminal length 0"};
2209         Task terminalLength = new Task("send", parametros0);
2210         Task observer = new Task("Observer");
2211
2212         String[] parametros = {"auto secure"};
2213         Task write = new Task("write", parametros);
2214         reconectar.exe();
2215         terminalLength.exe();
2216         data.setReadingAll(true);
2217         data.addObserver(this);
2218
2219         observer.start();
2220         write.exe();
2221
2222
2223
2224         // autosecureParser();
2225     } catch (Exception e) {
2226         JOptionPane.showMessageDialog(pan_conexion, e);
2227     } // TODO add your handling code here:
2228 }
2229
2230 private void aaut_btn_enviarActionPerformed(java.awt.event.ActionEvent evt)
{
2231     String envio = aaut_txtf_message.getText();
2232     if (envio.equals("")) {
2233         JOptionPane.showMessageDialog(pan_conexion, "Envio vacio.");

```

```

2234     } else {
2235         String[] parametros = {envio};
2236         Task write = new Task("write", parametros);
2237         try {
2238             write.exe();
2239         } catch (Exception e) {
2240             JOptionPane.showMessageDialog(pan_conexion, e.getMessage());
2241         }
2242         aaut_txtf_message.setText("");
2243     }
2244 }
2245 }
2246
2247 private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) {
2248     data.setReadingAll(false);
2249     data.deleteObserver(this);
2250     try {
2251         String[] parametros = {ETX + ""};
2252         Task write = new Task("write", parametros);
2253         Task desconectar = new Task("desconectar");
2254         write.exe();
2255         desconectar.exe();
2256     } catch (Exception e) {
2257         // JOptionPane.showMessageDialog(pan_conexion, e
2258         JOptionPane.showMessageDialog(pan_auto2, e.getMessage() + "error al
desconectar");
2259     }
2260 }
2261
2262 /*
2263
2264 private void autosecureParser() {
2265     String informacion = data.getBuffer();
2266     //AS_btn_start.setVisible(false);
2267
2268
2269
2270     AS_lbl_internetQ.setVisible(false);
2271     AS_rbtn_internetSi.setVisible(false);
2272     AS_rbtn_internetno.setVisible(false);
2273
2274     AS_lbl_numerointerfaces.setVisible(false);
2275     AS_spn_numerointerfaces.setVisible(false);
2276
2277     AS_scrpnl_interfaces.setVisible(false);
2278     AS_lbl_interfaces.setVisible(false);
2279
2280     AS_lbl_SNMP.setVisible(false);
2281     AS_rbtn_SNMPSi.setVisible(false);
2282     AS_rbtn_SNMPNo.setVisible(false);

```

```
2283
2284 AS_lbl_bienvenida.setVisible(false);
2285 AS_scrpnl_bienvenida.setVisible(false);
2286
2287 AS_lbl_secret.setVisible(false);
2288 AS_txt_secret.setVisible(false);
2289
2290 AS_lbl_password.setVisible(false);
2291 AS_txt_password.setVisible(false);
2292
2293 AS_lbl_ataques.setVisible(false);
2294 AS_spn_ataques.setVisible(false);
2295
2296 AS_lbl_intentos.setVisible(false);
2297 AS_spn_intentos.setVisible(false);
2298
2299 AS_lbl_tiempo.setVisible(false);
2300 AS_spn_tiempo.setVisible(false);
2301
2302 AS_lbl_SSH.setVisible(false);
2303 AS_rbtn_SSHSi.setVisible(false);
2304 AS_rbtn_SSHNo.setVisible(false);
2305
2306 AS_lbl_host.setVisible(false);
2307 AS_txt_host.setVisible(false);
2308
2309 AS_lbl_dominio.setVisible(false);
2310 AS_txt_dominio.setVisible(false);
2311
2312 AS_lbl_firewall.setVisible(false);
2313 AS_rbtn_firewalSi.setVisible(false);
2314 AS_rbtn_firewallNo.setVisible(false);
2315
2316
2317 AS_lbl_aplicar.setVisible(false);
2318 AS_rbtn_aplicarSi.setVisible(false);
2319 AS_rbtn_AplicarNo.setVisible(false);
2320
2321 AS_btn_enviarInfo.setVisible(true);
2322
2323 if (informacion.contains("Is this router connected to internet?")) {
2324
2325 AS_lbl_internetQ.setVisible(true);
2326 AS_rbtn_internetSi.setVisible(true);
2327 AS_rbtn_internetno.setVisible(true);
2328 autoSecureStep = 1;
2329 } else if (informacion.contains("Enter the number of interfaces facing the
internet")) {
2330 AS_lbl_numerointerfaces.setVisible(true);
2331 AS_spn_numerointerfaces.setVisible(true);
```

```

2332 autoSecureStep = 2;
2333 } else if (informacion.contains("Enter the interface name that is facing the
internet")) {
2334 String info = data.getBuffer();
2335 //Parsear el regreso para mostrarlo en la pantalla
2336
2337 String[] lines = info.split("\r\n");
2338 int linesSize = lines.length;
2339 String[][] tabla = new String[linesSize - 3][6];
2340 for (int i = 2; i <= linesSize - 2; i++) {
2341 String[] allWords = lines[i].split(" ");
2342 String[] words = new String[7];
2343
2344 int allWordsLegth = allWords.length;
2345 int temp = 0;
2346 for (int k = 0; k < allWordsLegth; k++) {
2347 if (!allWords[k].equals("")) {
2348 words[temp] = allWords[k];
2349 temp++;
2350 }
2351 }
2352
2353 int numWords = words.length;
2354 if (words[6] == null) {
2355
2356 for (int j = 0; j < 6; j++) {
2357 tabla[i - 2][j] = words[j];
2358 }
2359 } else {
2360 tabla[i - 2][0] = words[0];
2361 tabla[i - 2][1] = words[1];
2362 tabla[i - 2][2] = words[2];
2363 tabla[i - 2][3] = words[3];
2364 tabla[i - 2][4] = words[4] + " " + words[5];
2365 tabla[i - 2][5] = words[6];
2366
2367 }
2368 }
2369 //Llenar el contenido de la tabla
2370 String[] titulos = {
2371 "Interface", "IP-Address", "OK", "Method", "Status", "Protocol"
2372 };
2373 Object[][] contenido = tabla;
2374 DefaultTableModel tablaModelo = new DefaultTableModel();
2375 tablaModelo = new DefaultTableModel(contenido, titulos);
2376 AS_tbl_interfaces.setModel(tablaModelo);
2377
2378 AS_scrpnl_interfaces.setVisible(true);
2379 AS_lbl_interfaces.setVisible(true);
2380 autoSecureStep = 3;

```

```
2381 } else if (informacion.contains("Is SNMP used to manage the router?")) {
2382 AS_lbl_SNMP.setVisible(true);
2383 AS_rbtn_SNMPSi.setVisible(true);
2384 AS_rbtn_SNMPNo.setVisible(true);
2385 autoSecureStep = 4;
2386 } else if (informacion.contains("Enter the security banner")) {
2387 AS_lbl_bienvenida.setVisible(true);
2388 AS_scrpnl_bienvenida.setVisible(true);
2389 autoSecureStep = 5;
2390 } else if (informacion.contains("Enter the new enable secret")) {
2391 AS_lbl_secret.setText("Contraseña secret");
2392 AS_lbl_secret.setVisible(true);
2393 AS_txt_secret.setVisible(true);
2394 autoSecureStep = 6;
2395 } else if (informacion.contains("Confirm the enable secret")) {
2396 AS_lbl_secret.setText("Confirma la contraseña secret:");
2397 AS_txt_secret.setText("");
2398 AS_lbl_secret.setVisible(true);
2399 AS_txt_secret.setVisible(true);
2400 autoSecureStep = 7;
2401 } else if (informacion.contains("Enter the new enable password")) {
2402 AS_lbl_password.setText("Contraseña password");
2403 AS_lbl_password.setVisible(true);
2404 AS_txt_password.setVisible(true);
2405 autoSecureStep = 8;
2406 } else if (informacion.contains("Confirm the enable password")) {
2407 AS_lbl_password.setText("Confirma la contraseña password");
2408 AS_txt_password.setText("");
2409 AS_lbl_password.setVisible(true);
2410 AS_txt_password.setVisible(true);
2411 autoSecureStep = 9;
2412 } else if (informacion.contains("Blocking Period when Login Attack
detected:")) {
2413 AS_lbl_ataques.setVisible(true);
2414 AS_spn_ataques.setVisible(true);
2415 autoSecureStep = 10;
2416 } else if (informacion.contains("Maximum Login failures with the device:")) {
2417 AS_lbl_intentos.setVisible(true);
2418 AS_spn_intentos.setVisible(true);
2419 autoSecureStep = 11;
2420 } else if (informacion.contains("Maximum time period for crossing the failed
login attempts:")) {
2421 AS_lbl_tiempo.setVisible(true);
2422 AS_spn_tiempo.setVisible(true);
2423 autoSecureStep = 12;
2424 } else if (informacion.contains("Configure SSH server?")) {
2425 AS_lbl_SSH.setVisible(true);
2426 AS_rbtn_SSHSi.setVisible(true);
2427 AS_rbtn_SSHNo.setVisible(true);
2428 autoSecureStep = 13;
```

```

2429 } else if (informacion.contains("Enter the hostname")) {
2430 AS_lbl_host.setVisible(true);
2431 AS_txt_host.setVisible(true);
2432 autoSecureStep = 14;
2433 } else if (informacion.contains("Enter the domain-name")) {
2434 AS_lbl_dominio.setVisible(true);
2435 AS_txt_dominio.setVisible(true);
2436 autoSecureStep = 15;
2437 } else if (informacion.contains("Configure CBAC Firewall feature?")) {
2438 AS_lbl_firewall.setVisible(true);
2439 AS_rbtn_firewalSi.setVisible(true);
2440 AS_rbtn_firewallNo.setVisible(true);
2441 autoSecureStep = 16;
2442 } else if (informacion.contains("Apply this configuration to running-config? "))
{
2443 AS_lbl_aplicar.setVisible(true);
2444 AS_rbtn_aplicarSi.setVisible(true);
2445 AS_rbtn_AplicarNo.setVisible(true);
2446 autoSecureStep = 17;
2447
2448 } else if (informacion.contains("#")) {
2449 Task desconectar = new Task("desconectar");
2450
2451 try {
2452 desconectar.exe();
2453 AS_lbl_aplicar.setVisible(false);
2454 AS_rbtn_aplicarSi.setVisible(false);
2455 AS_rbtn_AplicarNo.setVisible(false);
2456 AS_btn_enviarInfo.setVisible(false);
2457 } catch (Exception e) {
2458 JOptionPane.showMessageDialog(pan_conexion, e + "Error ");
2459 }
2460 }
2461
2462 else {
2463 String[] blank = {" "};
2464 Task sendBlank = new Task("send", blank);
2465 try {
2466 bufferTemp = data.getBuffer();
2467 sendBlank.exe();
2468 data.setBuffer(bufferTemp);
2469 autosecureParser();
2470 AS_btn_enviarInfo.setVisible(false);
2471 } catch (Exception e) {
2472 JOptionPane.showMessageDialog(pan_conexion, e + " Error al enviar blank.");
2473 }
2474 }
2475 }
2476 */
2477 private void actualizaListaTareas(Task tarea) {

```

```

2478
2479     listaDeTareas.add(listaDeTareas.size() - 2, tarea);
2480     // listaDeTareas.add(tarea);
2481
2482     listaModelo.add(listaModelo.getSize() - 2, tarea.getTask());
2483     //listaModelo.addElement(tarea.getTask());
2484
2485     lst_tareas.setModel(listaModelo);
2486
2487     pgrb_ejecutar.setMaximum(listaDeTareas.size());
2488
2489 }
2490
2491 private void load() {
2492     boolean loadShowRunOK = false;
2493
2494     Task reconectar = new Task("reconectar");
2495     Task guardar = new Task("guardar");
2496     Task desconectar = new Task("desconectar");
2497
2498     listaDeTareas.add(reconectar);
2499     listaDeTareas.add(guardar);
2500     listaDeTareas.add(desconectar);
2501
2502     listaModelo.addElement(reconectar.getTask());
2503     listaModelo.addElement(guardar.getTask());
2504     listaModelo.addElement(desconectar.getTask());
2505     lst_tareas.setModel(listaModelo);
2506     numeroDeTareas = 3;
2507     pgrb_ejecutar.setMaximum(numeroDeTareas);
2508     pgrb_ejecutar.setValue(0);
2509     lst_tareas.ensureIndexIsVisible(lst_tareas.getModel().getSize() - 1);
2510
2511     try {
2512         reconectar.exe();
2513         Task loadShowRun = new Task("loadShowRun");
2514         loadShowRunOK = loadShowRun.exe();
2515
2516         if (!loadShowRunOK) {
2517             JOptionPane.showMessageDialog(pan_lineas, "Error al cargar
información");
2518
2519             } //Task desconectar = new Task("Desconectar");
2520             desconectar.exe();
2521
2522         } catch (Exception e) {
2523             JOptionPane.showMessageDialog(pan_lineas, e);
2524
2525         }
2526         if (loadShowRunOK) {

```

```

2527
2528     fillComponents();
2529
2530 }
2531
2532 }
2533
2534 private void fillComponents() {
2535     fillSecretPanel();
2536     fillLineasPanel();
2537     fillUsuariosPanel();
2538     fillConexionsPanel();
2539     fillAccesoPanel();
2540     fillNivelesPanel();
2541     fillPrivilegiosPanel();
2542     // fillAutoSecurePanel();
2543     fillMensajesPanel();
2544     fillNTPPanel();
2545     fillVistas();
2546     fillACL();
2547     fillSNMP();
2548
2549
2550
2551 }
2552
2553 private void fillLineasPanel() {
2554
2555     //Tabla
2556     showRunInfo.lineInfo();
2557     String[] titulos = {
2558         "Linea", "Contraseña"
2559     };
2560     Object[][] contenido = showRunInfo.lineInfo();
2561     DefaultTableModel tablaModelo = new DefaultTableModel();
2562     tablaModelo = new DefaultTableModel(contenido, titulos);
2563
2564     LIN_tbl_lineas.setModel(tablaModelo);
2565     //check box
2566
2567     DefaultComboBoxModel modeloComboBox = new
DefaultComboBoxModel(showRunInfo.lines());
2568     LIN_cbox_linea.setModel(modeloComboBox);
2569
2570 }
2571
2572 private void fillSecretPanel() {
2573     if (showRunInfo.passwordHabilitado()) {
2574         SCR_lbl_habilitado.setForeground(Color.GREEN);
2575         SCR_lbl_habilitado.setText("Habilitado");

```

```

2576     SCR_rbtn_deshabilita.setEnabled(true);
2577
2578     } else {
2579         SCR_lbl_habilitado.setForeground(Color.RED);
2580         SCR_lbl_habilitado.setText("Deshabilitado");
2581         SCR_rbtn_deshabilita.setEnabled(false);
2582
2583     }
2584 }
2585
2586 private void fillUsuariosPanel() {
2587     String[] titulos = {
2588         "Usuario", "Nivel", "Encriptación"
2589     };
2590     Object[][] contenido = showRunInfo.userInfo();
2591     DefaultTableModel tablaModelo = new DefaultTableModel();
2592
2593     tablaModelo = new DefaultTableModel(contenido, titulos);
2594     US_tbl_usuarios.setModel(tablaModelo);
2595
2596     US_tbl_usuarios.setSelectionMode(ListSelectionModel.SINGLE_SELECTION);
2597 }
2598
2599 private void fillConexionsPanel() {
2600     String[] titulos = {
2601         "Linea", "Minutos", "segundos"
2602     };
2603     Object[][] contenido = showRunInfo.conexionInfo();
2604     DefaultTableModel tablaModelo = new DefaultTableModel();
2605
2606     tablaModelo = new DefaultTableModel(contenido, titulos);
2607     CON_tbl_tiempos.setModel(tablaModelo);
2608
2609     CON_tbl_tiempos.setSelectionMode(ListSelectionModel.SINGLE_SELECTION);
2610
2611     DefaultComboBoxModel modeloComboBox = new
2612     DefaultComboBoxModel(showRunInfo.lines());
2613     CON_cbox_lineas.setModel(modeloComboBox);
2614 }
2615 private void fillAccesoPanel() {
2616     ACC_txt_numeros.setText(showRunInfo.intentosDeConexion());
2617 }
2618
2619 private void fillNivelesPanel() {
2620
2621     String[] titulos = {
2622         "Niveles", "Secreto"

```

```
2623     };
2624
2625     Object[][] contenido = showRunInfo.nivelesInfo();
2626     DefaultTableModel tablaModelo = new DefaultTableModel(contenido,
titulos);
2627
2628     NIV_tbl_niveles.setModel(tablaModelo);
2629
2630
2631
2632
2633 }
2634
2635 private void fillPrivilegiosPanel() {
2636
2637     String[] titulos = {
2638         "Comando", "Nivel", "Alcance"
2639     };
2640
2641     Object[][] contenido = showRunInfo.privilegiosInfo();
2642     DefaultTableModel tablaModelo = new DefaultTableModel(contenido,
titulos);
2643
2644     PRI_tbl_privilegios.setModel(tablaModelo);
2645
2646
2647
2648
2649 }
2650 /*
2651 private void fillAutoSecurePanel() {
2652
2653
2654
2655     AS_lbl_internetQ.setVisible(false);
2656     AS_rbtn_internetSi.setVisible(false);
2657     AS_rbtn_internetno.setVisible(false);
2658
2659     AS_lbl_numerointerfaces.setVisible(false);
2660     AS_spn_numerointerfaces.setVisible(false);
2661
2662     AS_scrpnl_interfaces.setVisible(false);
2663     AS_lbl_interfaces.setVisible(false);
2664
2665     AS_lbl_SNMP.setVisible(false);
2666     AS_rbtn_SNMPNo.setVisible(false);
2667     AS_rbtn_SNMPNo.setVisible(false);
2668
2669     AS_lbl_bienvenida.setVisible(false);
2670     AS_scrpnl_bienvenida.setVisible(false);
```

```
2671
2672 AS_lbl_secret.setVisible(false);
2673 AS_txt_secret.setVisible(false);
2674
2675 AS_lbl_password.setVisible(false);
2676 AS_txt_password.setVisible(false);
2677
2678 AS_lbl_ataques.setVisible(false);
2679 AS_spn_ataques.setVisible(false);
2680
2681 AS_lbl_intentos.setVisible(false);
2682 AS_spn_intentos.setVisible(false);
2683
2684 AS_lbl_tiempo.setVisible(false);
2685 AS_spn_tiempo.setVisible(false);
2686
2687 AS_lbl_SSH.setVisible(false);
2688 AS_rbtn_SSHSi.setVisible(false);
2689 AS_rbtn_SSHNo.setVisible(false);
2690
2691 AS_lbl_host.setVisible(false);
2692 AS_txt_host.setVisible(false);
2693
2694 AS_lbl_dominio.setVisible(false);
2695 AS_txt_dominio.setVisible(false);
2696
2697 AS_lbl_firewall.setVisible(false);
2698 AS_rbtn_firewalSi.setVisible(false);
2699 AS_rbtn_firewallNo.setVisible(false);
2700
2701
2702 AS_lbl_aplicar.setVisible(false);
2703 AS_rbtn_aplicarSi.setVisible(false);
2704 AS_rbtn_AplicarNo.setVisible(false);
2705
2706 AS_btn_enviarInfo.setVisible(false);
2707
2708
2709 }*/
2710
2711 private void fillMensajesPanel() {
2712     String[] titulos = {
2713         "Fecha", "Hora", "Evento"
2714     };
2715
2716     Object[][] contenido = new Object[0][0];
2717     DefaultTableModel tablaModelo = new DefaultTableModel(contenido,
titulos);
2718     MNS_tbl_mensajes.setModel(tablaModelo);
2719
```

```

2720 }
2721
2722 private void fillNTPPanel() {
2723
2724     //Invisibles temporales
2725     ntp_check_key.setVisible(false);
2726     ntp_lbl_numeroLlave.setVisible(false);
2727     ntp_txf_numeroLlave.setVisible(false);
2728     ntp_lbl_valorLlave.setVisible(false);
2729     ntp_txf_valorLlave.setVisible(false);
2730     //Llenado de tabla
2731     String[] titulos = {
2732         "Tipo de servicio", "Nombre o ip", "Preferido"
2733     };
2734     Object[][] contenido = showRunInfo.ntpInfo();
2735     DefaultTableModel tablaModelo = new DefaultTableModel();
2736
2737     tablaModelo = new DefaultTableModel(contenido, titulos);
2738     ntp_tbl_servidores.setModel(tablaModelo);
2739
ntp_tbl_servidores.setSelectionMode(ListSelectionModel.SINGLE_SELECTION);
2740
2741 }
2742
2743 private void fillVistas() {
2744
2745     //Llenado de tabla
2746     String[] titulos = {
2747         "Nombre", "MIB", "Condición"
2748     };
2749     Object[][] contenido = showRunInfo.vistasDetailedInfo();
2750     DefaultTableModel tablaModelo = new DefaultTableModel();
2751
2752     tablaModelo = new DefaultTableModel(contenido, titulos);
2753     vis_tbl_info.setModel(tablaModelo);
2754     vis_tbl_info.setSelectionMode(ListSelectionModel.SINGLE_SELECTION);
2755
2756 }
2757
2758 private void fillACL() {
2759     //Llenado de tabla
2760     String[] titulos = {
2761         "Lista", "Estado", "Anfitrión"
2762     };
2763     Object[][] contenido = showRunInfo.accessListDetailedInfo();
2764     DefaultTableModel tablaModelo = new DefaultTableModel();
2765
2766     tablaModelo = new DefaultTableModel(contenido, titulos);
2767     acl_tbl_info.setModel(tablaModelo);
2768     acl_tbl_info.setSelectionMode(ListSelectionModel.SINGLE_SELECTION);

```

```

2769 }
2770
2771 private void fillSNMP() {
2772
2773     //Llenado de tabla
2774     String[] titulos = {
2775         "Comunidad", "Vista", "Tipo", "ACL"
2776     };
2777     Object[][] contenido = showRunInfo.anmpServerCommunityInfo();
2778     DefaultTableModel tablaModelo = new DefaultTableModel();
2779
2780     tablaModelo = new DefaultTableModel(contenido, titulos);
2781     snmp_tbl_info.setModel(tablaModelo);
2782
2783     snmp_tbl_info.setSelectionMode(ListSelectionModel.SINGLE_SELECTION);
2784
2785     DefaultComboBoxModel modeloComboBox = new
2786     DefaultComboBoxModel(showRunInfo.aclList());
2787     snmp_cbox_listas.setModel(modeloComboBox);
2788     modeloComboBox = new
2789     DefaultComboBoxModel(showRunInfo.vistasList());
2790     snmp_cbox_vistas.setModel(modeloComboBox);
2791 }
2792 /**
2793  * @param args the command line arguments
2794  */
2795 public static void main(String args[]) {
2796     java.awt.EventQueue.invokeLater(new Runnable() {
2797
2798         public void run() {
2799             new MainGUI().setVisible(true);
2800
2801
2802
2803
2804         }
2805     });
2806
2807
2808
2809
2810 }
2811 // Variables declaration - do not modify
2812 private javax.swing.JButton ACC_btn_anadir;
2813 private javax.swing.JLabel ACC_lbl_intentos;
2814 private javax.swing.JSpinner ACC_spn_intentos;
2815 private javax.swing.JTextField ACC_txt_numeros;

```

```
2816 private javax.swing.ButtonGroup AS_btngrp_SNMP;
2817 private javax.swing.ButtonGroup AS_btngrp_SSH;
2818 private javax.swing.ButtonGroup AS_btngrp_aplicar;
2819 private javax.swing.ButtonGroup AS_btngrp_firewall;
2820 private javax.swing.ButtonGroup AS_btngrp_internet;
2821 private javax.swing.JButton CON_bytn_aniadir;
2822 private javax.swing.JComboBox CON_cbox_lineas;
2823 private javax.swing.JLabel CON_lbl_linea;
2824 private javax.swing.JLabel CON_lbl_minutos;
2825 private javax.swing.JLabel CON_lbl_segundos;
2826 private javax.swing.JScrollPane CON_scrpnl_conexiones;
2827 private javax.swing.JSpinner CON_spin_min;
2828 private javax.swing.JSpinner CON_spin_sec;
2829 private javax.swing.JTable CON_tbl_tiempos;
2830 private javax.swing.JLabel IG_lbl_HostName;
2831 private javax.swing.JLabel IG_lbl_dispositivo;
2832 private javax.swing.JScrollPane IG_scr_showTechSupport;
2833 private javax.swing.JTextArea IG_txta_showTechSupport;
2834 private javax.swing.JButton LIN_btn_aniadir;
2835 private javax.swing.JComboBox LIN_cbox_linea;
2836 private javax.swing.JLabel LIN_lbl_contrasenia;
2837 private javax.swing.JLabel LIN_lbl_lineas;
2838 private javax.swing.JScrollPane LIN_scrpnl_lineas;
2839 private javax.swing.JTable LIN_tbl_lineas;
2840 private javax.swing.JTextField LIN_txt_nContrasenia;
2841 private javax.swing.JButton MNS_btn_comenzar;
2842 private javax.swing.JButton MNS_btn_prioridad;
2843 private javax.swing.JButton MNS_btn_terminal;
2844 private javax.swing.JButton MNS_btn_terminar;
2845 private javax.swing.JComboBox MNS_cbox_prioridad;
2846 private javax.swing.JLabel MNS_lbl_prioridad;
2847 private javax.swing.JLabel MNS_lbl_propiedades;
2848 private javax.swing.JLabel MNS_lbl_terminal;
2849 private javax.swing.JLabel MNS_lbl_ver;
2850 private javax.swing.JScrollPane MNS_scrpnl_mensajes;
2851 private javax.swing.JSpinner MNS_spn_terminal;
2852 private javax.swing.JTable MNS_tbl_mensajes;
2853 private javax.swing.JButton NIV_btn_aniadir;
2854 private javax.swing.JLabel NIV_lbl_contrasenia;
2855 private javax.swing.JScrollPane NIV_scrpnl_niveles;
2856 private javax.swing.JTable NIV_tbl_niveles;
2857 private javax.swing.JTextField NIV_txt_nContrasenia;
2858 private javax.swing.JButton PRI_btn_aniadir;
2859 private javax.swing.JButton PRI_btn_opciones;
2860 private javax.swing.ButtonGroup PRI_btngrp_opciones;
2861 private javax.swing.JCheckBox PRI_chkbtn_all;
2862 private javax.swing.JCheckBox PRI_chkbtn_reset;
2863 private javax.swing.JLabel PRI_lbl_nivel;
2864 private javax.swing.JList PRI_lst_opciones;
2865 private javax.swing.JScrollPane PRI_scrpnl_privilegios;
```

```
2866 private javax.swing.JSpinner PRI_spn_nivel;
2867 private javax.swing.JTable PRI_tbl_privilegios;
2868 private javax.swing.JTextField PRI_txt_comando;
2869 private javax.swing.ButtonGroup SCR_btn_habilitar;
2870 private javax.swing.JLabel SCR_lbl_habilitado;
2871 private javax.swing.JLabel SCR_lbl_secret;
2872 private javax.swing.JLabel SCR_lbl_solicitud;
2873 private javax.swing.JLabel SCR_lbl_texto;
2874 private javax.swing.JRadioButton SCR_rbtn_deshabilitar;
2875 private javax.swing.JRadioButton SCR_rbtn_habilitar;
2876 private javax.swing.JTextField SCR_txt_SecretPassword;
2877 private javax.swing.JButton SCT_btn_aniadir;
2878 private javax.swing.JButton US_btn_aniadir;
2879 private javax.swing.JButton US_btn_eliminar;
2880 private javax.swing.JComboBox US_cbox_encryption;
2881 private javax.swing.JLabel US_lbl_contrasenia;
2882 private javax.swing.JLabel US_lbl_encryption;
2883 private javax.swing.JLabel US_lbl_nivelPrivilegios;
2884 private javax.swing.JLabel US_lbl_noPrivilegio;
2885 private javax.swing.JLabel US_lbl_usuario;
2886 private javax.swing.JScrollPane US_scrpnl_usuarios;
2887 private javax.swing.JSlider US_sld_privilegios;
2888 private javax.swing.JTable US_tbl_usuarios;
2889 private javax.swing.JTextField US_txt_contrasenia;
2890 private javax.swing.JTextField US_txt_usuario;
2891 private javax.swing.JButton aaut_btn_comenzar;
2892 private javax.swing.JButton aaut_btn_enviar;
2893 private javax.swing.JScrollPane aaut_scrpnl_info;
2894 private javax.swing.JTextField aaut_txtf_message;
2895 private javax.swing.JTextArea aauto_txta_info;
2896 private javax.swing.JButton acl_btn_aniadir;
2897 private javax.swing.JRadioButton acl_btn_deny;
2898 private javax.swing.JButton acl_btn_eliminar;
2899 private javax.swing.JRadioButton acl_btn_permit;
2900 private javax.swing.ButtonGroup acl_btngrp_acl;
2901 private javax.swing.JLabel acl_lbl_acl;
2902 private javax.swing.JLabel acl_lbl_host;
2903 private javax.swing.JLabel acl_lbl_numeralista;
2904 private javax.swing.JScrollPane acl_scrpnl_info;
2905 private javax.swing.JSpinner acl_spn_numeralista;
2906 private javax.swing.JTable acl_tbl_info;
2907 private javax.swing.JTextField acl_txtf_host;
2908 private javax.swing.JButton btn_ShowTechSupport;
2909 private javax.swing.JButton btn_desconectar;
2910 private javax.swing.JButton btn_ejecutar;
2911 private javax.swing.JButton jButton1;
2912 private javax.swing.JButton jButton2;
2913 private javax.swing.JButton jButton3;
2914 private javax.swing.JScrollPane jScrollPane1;
2915 private javax.swing.JScrollPane jScrollPane10;
```

```
2916 private javax.swing.JScrollPane jScrollPane2;
2917 private javax.swing.JLabel lbl_tareas;
2918 private javax.swing.JList lst_tareas;
2919 private javax.swing.JButton ntp_btn_aniadir;
2920 private javax.swing.JButton ntp_btn_eliminar;
2921 private javax.swing.JComboBox ntp_cbox_tipoServicio;
2922 private javax.swing.JCheckBox ntp_check_key;
2923 private javax.swing.JCheckBox ntp_check_prefered;
2924 private javax.swing.JLabel ntp_lbl_nombreServicio;
2925 private javax.swing.JLabel ntp_lbl_numeroLlave;
2926 private javax.swing.JLabel ntp_lbl_tipoServicio;
2927 private javax.swing.JLabel ntp_lbl_valorLlave;
2928 private javax.swing.JScrollPane ntp_scrpnl_servidores;
2929 private javax.swing.JTable ntp_tbl_servidores;
2930 private javax.swing.JTextField ntp_txtf_nombreServicio;
2931 private javax.swing.JTextField ntp_txtf_numeroLlave;
2932 private javax.swing.JTextField ntp_txtf_valorLlave;
2933 private javax.swing.JPanel pan_InformacionGeneral;
2934 private javax.swing.JPanel pan_acceso;
2935 private javax.swing.JPanel pan_acl;
2936 private javax.swing.JPanel pan_auto2;
2937 private javax.swing.JPanel pan_conexion;
2938 private javax.swing.JPanel pan_lineas;
2939 private javax.swing.JPanel pan_mensajes;
2940 private javax.swing.JPanel pan_niveles;
2941 private javax.swing.JPanel pan_ntp;
2942 private javax.swing.JPanel pan_privilegios;
2943 private javax.swing.JPanel pan_secret;
2944 private javax.swing.JPanel pan_ssh;
2945 private javax.swing.JPanel pan_usuarios;
2946 private javax.swing.JPanel pan_vistas;
2947 private javax.swing.JProgressBar pgrb_ejecutar;
2948 private javax.swing.JButton snmp_btn_aniadir;
2949 private javax.swing.JButton snmp_btn_eliminar;
2950 private javax.swing.JComboBox snmp_cbox_listas;
2951 private javax.swing.JComboBox snmp_cbox_tipo;
2952 private javax.swing.JComboBox snmp_cbox_vistas;
2953 private javax.swing.JPanel snmp_lbl_comunidades;
2954 private javax.swing.JLabel snmp_lbl_listas;
2955 private javax.swing.JLabel snmp_lbl_nombre;
2956 private javax.swing.JLabel snmp_lbl_tipo;
2957 private javax.swing.JLabel snmp_lbl_titulo;
2958 private javax.swing.JLabel snmp_lbl_vistas;
2959 private javax.swing.JScrollPane snmp_scrpnl_info;
2960 private javax.swing.JTable snmp_tbl_info;
2961 private javax.swing.JTextField snmp_txtf_nombre;
2962 private javax.swing.JButton ssh_btn_aniadir;
2963 private javax.swing.JCheckBox ssh_check_nuevaLlave;
2964 private javax.swing.JLabel ssh_lbl_RSA;
2965 private javax.swing.JLabel ssh_lbl_anfitrion;
```

```
2966 private javax.swing.JLabel ssh_lbl_dominio;
2967 private javax.swing.JLabel ssh_lbl_modulo;
2968 private javax.swing.JScrollPane ssh_scrpnl_llaves;
2969 private javax.swing.JSpinner ssh_spn_tamano;
2970 private javax.swing.JTextArea ssh_txta_llaves;
2971 private javax.swing.JTextField ssh_txtf_anfitrion;
2972 private javax.swing.JTextField ssh_txtf_dominio;
2973 private javax.swing.JTabbedPane tab_principal;
2974 private javax.swing.JButton vis_btn_aniadir;
2975 private javax.swing.JButton vis_btn_eliminar;
2976 private javax.swing.JRadioButton vis_btn_excluded;
2977 private javax.swing.JRadioButton vis_btn_included;
2978 private javax.swing.ButtonGroup vis_btngrp_includedExcluded;
2979 private javax.swing.JLabel vis_lbl_mib;
2980 private javax.swing.JLabel vis_lbl_nombre;
2981 private javax.swing.JLabel vis_lbl_vistas;
2982 private javax.swing.JTable vis_tbl_info;
2983 private javax.swing.JTextField vis_txtf_MIB;
2984 private javax.swing.JTextField vis_txtf_nombre;
2985 // End of variables declaration
2986 }
```

## ***Paquete SNMPLogic***

### **Clase Main**

```
01 /*
02 * To change this template, choose Tools | Templates
03 * and open the template in the editor.
04 */
05 package snmplogic;
06
07 import GUI.TelnetGUI;
08 import GUI.LoginGUI;
09 //import org.apache.commons.net.telnet.TelnetClient;
10
11 /**
12 *
13 * @author RAM
14 */
15 public class Main {
16
17     /**
18     * @param args the command line arguments
19     */
20     public static void main(String[] args) {
21         java.awt.EventQueue.invokeLater(new Runnable() {
22             public void run() {
23                 new LoginGUI().setVisible(true);
24             }
25         });
26         // TODO code application logic here
27     }
28 }
```

## Clase ShowRunParser

```
001 /*
002  * To change this template, choose Tools | Templates
003  * and open the template in the editor.
004  */
005 package snmplogic;
006
007 import snmplogic.almacenamiento.Data;
008 import java.util.ArrayList;
009 import java.util.List;
010 import java.util.regex.Matcher;
011 import java.util.regex.Pattern;
012
013 /**
014  *
015  * @author RAM
016  */
017 public class ShowRunParser {
018
019     Data data = new Data();
020
021     public ShowRunParser() {
022         this.data = data.getInstance();
023     }
024
025     public boolean findLineStartsWith(String text) {
026         String[] lines = data.getShowRun().split("\r\n");
027         boolean encontrado = false;
028         for (int i = 0; i < lines.length; i++) {
029             if (lines[i].startsWith(text)) {
030                 encontrado = true;
031             }
032         }
033     }
034     return encontrado;
035 }
036
037 public boolean findOneLineWithPattern (String strPattern){
038     boolean retorno = false;
039     String [] temp = getAllLinesInText(strPattern);
040     if (temp.length == 1){
041         retorno = true;
042     }
043     return retorno;
044 }
045
046 public String[] findAlllinesStartWith (String strStart){
```

```

047     String [] textInLines = data.getShowRun().split("\r\n");
048     int textSize = textInLines.length;
049     List<String> listaTemp = new ArrayList<String>();
050
051     for (int i = 0; i < textSize; i++){
052         if (textInLines[i].startsWith(strStart)){
053             listaTemp.add(textInLines[i]);
054         }
055     }
056     return listaTemp.toArray(new String[0]);
057 }
058
059 public String[] getDataFromNextLineAfterReference(String referencia, String
patron) {
060     String[] lines = new String[0];
061
062     lines = getLinesReferenceAndPattern(data.getShowRun(), referencia, patron);
063     if (lines.length == 2) {
064         return getDataFromLine(lines[1], patron);
065     }
066     return null;
067 }
068
069 public String[] getAllLinesReferenceAndPattern(String referencia, String patron)
{
070     String[] lines = new String[0];
071
072     lines = getLinesReferenceAndPattern(data.getShowRun(), referencia, patron);
073
074     return lines;
075
076
077 }
078
079 public String[] getAllLinesInText(String strPattern){
080
081     return getLinesFromText(data.getShowRun(), strPattern);
082 }
083
084 public String[] getDataFirstLineInText(String patron) {
085     String[] lines = new String[0];
086
087     lines = getLinesFromText(data.getShowRun(), patron);
088     if (lines.length == 1) {
089         return getDataFromLine(lines[0], patron);
090     }
091     return null;
092 }
093

```

```

094 private String[] getLinesReferenceAndPattern(String strText, String
strReference, String strPattern) {
095
096     String strNormalizedPattern = normalizePattern(strPattern);
097     String strNormalizedReference = normalizePattern(strReference);
098
099     List<String> listaRetorno = new ArrayList<String>();
100
101     Pattern pattern = Pattern.compile(strNormalizedPattern);
102     Pattern reference = Pattern.compile(strNormalizedReference);
103
104     String[] textInLines = strText.split("\r\n");
105     for (int i = 0; i < textInLines.length; i++) {
106
107         Matcher matcherPattern = pattern.matcher(textInLines[i]);
108         matcherPattern.find();
109
110         Matcher matcherReference = reference.matcher(textInLines[i]);
111         matcherReference.find();
112         if (matcherReference.matches()) {
113             listaRetorno.add(textInLines[i]);
114         }
115         if (matcherPattern.matches()) {
116             listaRetorno.add(textInLines[i]);
117         }
118     }
119     return listaRetorno.toArray(new String[0]);
120 }
121
122 private String[] getLinesFromText(String strText, String strPattern) {
123
124     String strNormalizedPattern = normalizePattern(strPattern);
125
126     List<String> listaRetorno = new ArrayList<String>();
127     Pattern pattern = Pattern.compile(strNormalizedPattern);
128
129     String[] textInLines = strText.split("\r\n");
130     for (int i = 0; i < textInLines.length; i++) {
131         Matcher matcher = pattern.matcher(textInLines[i]);
132         matcher.find();
133         if (matcher.matches()) {
134             listaRetorno.add(textInLines[i]);
135         }
136     }
137     return listaRetorno.toArray(new String[0]);
138
139 }
140 }
141
142 public String[] getDataFromLine(String strLine, String strPattern) {

```

```

143
144     List<String> regreso = new ArrayList<String>();
145
146     String[] wordsInPattern = strPattern.split(" ");
147     String[] wordsInLine = strLine.split(" ");
148     for (int i = 0; i < wordsInLine.length; i++) {
149         if (wordsInPattern[i].startsWith("<")
150             && wordsInPattern[i].endsWith(">")) {
151             regreso.add(wordsInLine[i]);
152         }
153     }
154     return regreso.toArray(new String[0]);
155 }
156
157 private String normalizePattern(String strPattern) {
158     strPattern = strPattern.replaceAll("<StringOb>", "[a-zA-Z0-9]+");
159     strPattern = strPattern.replaceAll("<StringOp>", "[ ]?[a-zA-Z0-9]+?");
160     strPattern = strPattern.replaceAll("<intOb>", "[0-9]");
161     strPattern = strPattern.replaceAll("<intOp>", "[ ]?[0-9]?");
162     strPattern = strPattern.replaceAll("<largeintOb>", "[0-9]+");
163     strPattern = strPattern.replaceAll("<largeintOp>", "[ ]?[0-9]+?");
164     return strPattern ;
165 }
166 }

```

## ***Paquete SNMPLogic.almacenamiento***

### **Clase Data**

```
001 /*
002 * To change this template, choose Tools | Templates
003 * and open the template in the editor.
004 */
005 package snmplogic.almacenamiento;
006
007 import java.util.Observable;
008
009 /**
010 *
011 * @author RAM
012 */
013 public class Data extends Observable {
014
015
016     private static Data instance = null;
017     private String ip = null;
018     private String hostName = null;
019     private boolean telnet = false;
020     private boolean SSH = false;
021     private String username = null;
022     private String password = null;
023     private boolean aaaEnabled = false;
024     private String enableAdminPass = null;
025     private int loginReturn = 0;
026     private String showRun = null;
027     private String buffer = null;
028     private boolean conectado = false;
029     private String[] options = null;
030     private boolean reading = false;
031     private String messageBuffer = null;
032     private boolean readingAll = false;
033
034     public boolean isReadingAll() {
035         return readingAll;
036     }
037
038     public void setReadingAll(boolean readingAll) {
039         this.readingAll = readingAll;
040     }
041
042     public boolean setProtocol(int noProtocolo) {
043         boolean temp = false;
```

```
044     switch (noProtocolo) {
045         case 1:
046             setTelnet(true);
047             temp = true;
048             break;
049         case 2:
050             setSSH(true);
051             temp = true;
052             break;
053         default:
054             break;
055     }
056     return temp;
057
058 }
059
060 public String getPassword() {
061     return password;
062 }
063
064 public void setPassword(String password) {
065     this.password = password;
066 }
067
068 public String getUsername() {
069     return username;
070 }
071
072 public void setUsername(String username) {
073     this.username = username;
074 }
075
076 public boolean isSSH() {
077     return SSH;
078 }
079
080 public void setSSH(boolean SSH) {
081     if (SSH) {
082         this.SSH = SSH;
083         this.telnet = !SSH;
084     }
085 }
086
087 public boolean isTelnet() {
088     return telnet;
089 }
090
091 public void setTelnet(boolean telnet) {
092     if (telnet) {
093         this.telnet = telnet;
```

```
094     this.SSH = !telnet;
095   }
096 }
097
098 public String getHostName() {
099     return hostName;
100 }
101
102 public void setHostName(String hostName) {
103     this.hostName = hostName;
104 }
105
106 public String getIp() {
107     return ip;
108 }
109
110 public void setIp(String ip) {
111     this.ip = ip;
112 }
113
114 void Data() {
115 }
116
117 public Data getInstance() {
118     if (instance == null) {
119         instance = new Data();
120     }
121     return instance;
122 }
123
124 public boolean isAaaEnabled() {
125     return aaaEnabled;
126 }
127
128 public void setAaaEnabled(boolean aaaEnabled) {
129     this.aaaEnabled = aaaEnabled;
130 }
131
132 public String getEnableAdminPass() {
133     return enableAdminPass;
134 }
135
136 public void setEnableAdminPass(String enableAdminPass) {
137     this.enableAdminPass = enableAdminPass;
138 }
139
140 public String getBuffer() {
141     return buffer;
142 }
143
```

```
144 public void setBuffer(String buffer) {
145     this.buffer = buffer;
146 }
147
148 public int getLoginReturn() {
149     return loginReturn;
150 }
151
152 public void setLoginReturn(int loginReturn) {
153     this.loginReturn = loginReturn;
154 }
155
156 public String getShowRun() {
157     return showRun;
158 }
159
160 public void setShowRun(String showRun) {
161     this.showRun = showRun;
162 }
163
164 public boolean isConectado() {
165     return conectado;
166 }
167
168 public void setConectado(boolean conectado) {
169     this.conectado = conectado;
170 }
171
172 public String[] getOptions() {
173     return options;
174 }
175
176 public void setOptions(String[] options) {
177     this.options = options;
178 }
179
180 public boolean isReading() {
181     return reading;
182 }
183
184 public void setReading(boolean reading) {
185     this.reading = reading;
186     // setChanged();
187     // notifyObservers(reading);
188 }
189
190 public String getMessageBuffer() {
191     return messageBuffer;
192 }
193
```

```
194 public void setMessageBuffer(String messageBuffer) {
195     this.messageBuffer = messageBuffer;
196     setChanged();
197     notifyObservers(messageBuffer);
198
199 }
200 }
```

## ***Paquete SNMPlogic.interfaces***

### **Clase Protocolo**

```
01 /*
02 * To change this template, choose Tools | Templates
03 /*
04 * To change this template, choose Tools | Templates
05 * and open the template in the editor.
06 */
07 package snmplogic.interfaces;
08
09 import java.util.List;
10
11 /**
12 *
13 * @author RAM
14 */
15 public interface Protocolo {
16
17     public Protocolo getInstance();
18
19     public String login(String ip) throws Exception;
20
21     public boolean disconnect() throws Exception;
22
23     public String exe(String command, String readUntil) throws Exception;
24
25     public String exe(String command, String readUntil, int times) throws Exception;
26
27     public String getTimer();
28
29     public void readMessages() throws Exception;
30
31     public void readAll() throws Exception;
32
33     public void write(String command);
34
35     void keepConnected(long milisecons) throws Exception;
36
37     public String getLastReaduntil();
38
39     public void setLastReadUntil(String readUntil);
40 }
```

## ***Paquete SNMPlogic.protocolos***

### **Clase Telnet**

```
001 /*
002 * To change this template, choose Tools | Templates
003 * and open the template in the editor.
004 */
005 package snmplogic.protocolos;
006
007 /**
008 *
009 * @author RAM
010 */
011 import java.util.Timer;
012 import java.util.TimerTask;
013
014
015 import org.apache.commons.net.telnet.TelnetClient;
016
017 import java.io.InputStream;
018 import java.io.PrintStream;
019 import snmplogic.interfaces.Protocolo;
020 import snmplogic.almacenamiento.Data;
021
022 public class Telnet implements Protocolo {
023
024     //public static final char STX = (char) 0x02;
025     public static final char ETX = (char) 0x03;
026     private static Telnet instance = null;
027     private Timer timer = null;
028     private TelnetClient telnetClient = null;
029     private InputStream in = null;
030     private PrintStream out = null;
031     private String lastReaduntil = null;
032     private boolean Conexionsuccessful = false;
033     Data data = new Data();
034     Thread runner;
035
036     /**
037     *Constructor.
038     * @param void
039     *
040     */
041     public Telnet() {
042         this.data = data.getInstance();
043         //runner = new Thread(this);
```

```

044     //runner.start();
045 }
046
047 /**
048  * Regresa siempre la misma instancia aunque el objeto sea nuevo.
049  * Singleton.
050  * @param void
051  * @return Telnet Regresa siempre la misma instancia de Telnet.
052  */
053 @Override
054 public Telnet getInstance() {
055     if (instance == null) {
056         instance = new Telnet();
057     }
058     return instance;
059 }
060
061 /**
062  * Se conecta con la ip que se le pasa como parámetro la cadena de la ip en
063  * formato de IP ###.###.###.###
064  * @param ip String con la ip del
065  */
066 public void connect(String ip) throws Exception {
067
068     System.out.println("Conectando a " + ip);
069     try {
070         this.telnetClient = new TelnetClient();
071
072         this.telnetClient.connect(ip, 23);
073
074         this.Connectionsuccessful = true;
075
076
077     } catch (Exception e) {
078         throw new Exception(e.getMessage());
079     } finally {
080
081         if (this.Connectionsuccessful == true) {
082             in = telnetClient.getInputStream();
083             out = new PrintStream(telnetClient.getOutputStream());
084         }
085     }
086 }
087
088 /**
089  * Hace la conexión inicial y regresa la Lista de Strings inicial,
090  * espera hasta encontrar cualquiera de los siguientes caracteres: ':', '#'
091  * o '>'.
092  * Nota: Si en medio de una cadena de caracteres se encuentra cualquiera de
093  * estos caracteres, la lectura se detendrá, lo que se debe hacer es dar

```

```

094 * un espacio y un enter para que la lectura reinicie.
095 */
096 @Override
097 public String login(String ip) throws Exception {
098
099     String regreso = new String("");
100
101     try {
102         connect(ip);
103     } catch (Exception e) {
104
105         throw new Exception(e.getMessage());
106
107     }
108
109     regreso = readUntil("Complete");
110     System.out.println(regreso);
111     return regreso;
112
113 }
114
115 /**
116 *Desconecta el cliente telnet y cierra todos.
117 */
118 @Override
119 public boolean disconnect() throws Exception {
120     System.out.println("Desconectado.");
121     boolean result = false;
122     if (this.Connectionsuccessful) {
123         try {
124             // Liberamos recursos
125             out.close();
126             in.close();
127             telnetClient.disconnect();
128             this.Connectionsuccessful = false;
129             result = true;
130         } catch (Exception e) {
131             new Exception("Error al desconectar" + e.getMessage());
132         }
133     }
134     return result;
135 }
136
137 /**
138 *Regresa el Timer solo para cuestiones de prueba.
139 */
140 @Override
141 public String getTimer() {
142     // TODO Auto-generated method stub
143     return this.timer.toString();

```

```

144 }
145
146 /**
147  *Manda el comando al router y regresa la lectura de del router en forma de
148  * una lista de cadenas List<String>.
149  */
150 @Override
151 public String exe(String command, String streadUntil) throws Exception {
152
153     String regreso = new String("");
154     //Cuando se envia un ' ' al router para que no se pierda la conexión.
155     this.lastReaduntil = streadUntil;
156     try {
157
158         //regreso = regreso.concat(command + "\n");
159
160         write(command);
161
162         regreso = regreso.concat(readUntil(streadUntil));
163         System.out.println(regreso);
164
165     } catch (Exception e) {
166
167         throw new Exception("Error al enviar el commando: " + command + "->" +
168 e.getMessage());
169     }
170     return regreso;
171 }
172 @Override
173 public String exe(String command, String streadUntil, int times) throws
174 Exception {
175
176     String regreso = new String("");
177     //Cuando se envia un ' ' al router para que no se pierda la conexión.
178     this.lastReaduntil = streadUntil;
179     try {
180
181         //regreso = regreso.concat(command + "\n");
182
183         write(command);
184
185         for (int i = 1; i <= times; i++) {
186             regreso = regreso.concat(readUntil(streadUntil));
187         }
188
189
190         System.out.println(regreso);
191

```

```

192     } catch (Exception e) {
193
194         throw new Exception("Error al enviar el comando: " + command + "->" +
e.getMessage());
195     }
196     return regreso;
197 }
198
199 /**
200  *
201  */
202 /**
203  * Escribe el comando en la salida hacia el router.
204  */
205 public void write(String command) {
206     //Lanzamos el comando
207     try {
208         out.println(command);
209         out.flush();
210
211     } catch (Exception e) {
212         e.printStackTrace();
213     }
214
215
216 }
217
218 /**
219  * Lee el retorno del router, hasta una cadena especificada.
220  */
221 public String readUntil(String strreadUntil) {
222
223     boolean finGenerico = false;
224
225     if (strreadUntil.equals("Complete")) {
226         finGenerico = true;
227     }
228
229     //Cadena temporal
230     StringBuffer sb = new StringBuffer();
231
232     try {
233
234         char lastChar = strreadUntil.charAt(strreadUntil.length() - 1);
235         int check = in.read();
236         char ch = (char) check;
237         while (check != -1) {
238
239             //System.out.print(ch);
240             sb.append(ch);

```

```

241         if (finGenerico) {
242             if (ch == '>' || ch == '#' || ch == ':') {
243                 return sb.toString();
244             }
245         } else {
246             if (ch == lastChar) {
247                 if (sb.toString().endsWith(streadUntil)) {
248                     return sb.toString();
249                 }
250             }
251         }
252
253
254
255         check = in.read();
256         ch = (char) check;
257     }
258 } catch (Exception e) {
259     e.printStackTrace();
260 }
261
262 return sb.toString();
263 }
264
265 @Override
266 public void keepConnected(long milisecons) throws Exception {
267     TimerTask timerTask = new TimerTask() {
268
269         public void run() {
270             try {
271                 write(" ");
272             } catch (Exception e) {
273             }
274             if (data.isReading()) {
275                 System.out.println("Keeping conected :");
276             } else {
277                 System.out.println("Finishing keep conected");
278                 timer.cancel();
279             }
280         }
281     };
282     this.timer = new Timer();
283
284     timer.scheduleAtFixedRate(timerTask, milisecons, milisecons);
285
286
287 }
288
289 @Override
290 public String getLastReaduntil() {

```

```

291     return lastReaduntil;
292 }
293
294 public void setLastReadUntil(String readUntil) {
295     this.lastReaduntil = readUntil;
296 }
297
298 public void readMessages() throws Exception {
299     String lineaActual = new String();
300
301     try {
302
303         StringBuffer sb = new StringBuffer();
304         int check = in.read();
305         char ch = (char) check;
306
307         while (check != -1) {
308             sb.append(ch);
309             check = in.read();
310             ch = (char) check;
311             if (ch == '\n') {
312
313                 lineaActual = sb.toString().trim();
314
315                 if (lineaActual.startsWith("*")) {
316                     data.setMessageBuffer(lineaActual);
317                     System.out.println("Reading messages " + lineaActual);
318                 }
319                 if (!data.isReading()) {
320                     check = -1;
321                 }
322                 sb = new StringBuffer();
323             }
324         }
325         System.out.println("Finishing read messages");
326     } catch (Exception e) {
327         throw new Exception(e.getMessage());
328     }
329 }
330 }
331
332 public void readAll() throws Exception {
333     String lineaActual = new String();
334
335     try {
336
337         StringBuffer sb = new StringBuffer();
338         int check = in.read();
339         char ch = (char) check;
340

```

```

341     while (check != -1) {
342         sb.append(ch);
343         check = in.read();
344         ch = (char) check;
345         if (ch == '\n' || ch == ':') {
346
347             lineaActual = sb.toString().trim();
348
349             //if (lineaActual.startsWith("*")) {
350             data.setMessageBuffer(lineaActual + "\n");
351             System.out.println("Reading all " + lineaActual);
352             if (lineaActual.contains(data.getHostName() + "#")) {
353                 write(ETX + "");
354                 disconnect();
355             }
356             //}
357             if (!data.isReadingAll()) {
358                 check = -1;
359             }
360             sb = new StringBuffer();
361         }
362     }
363     System.out.println("Finishing read all");
364 } catch (Exception e) {
365     throw new Exception(e.getMessage());
366 }
367
368 }
369
370 /* @Override
371 public void run() {
372     try {
373         keepConnected(30000L);
374         readMessages();
375     } catch (Exception e) {
376         System.out.println(e);
377     }
378 }*/
379 }

```

## ***Paquete SNMPlogic.tareas***

### **Clase ShowRunInfo**

```
001 /*
002 * To change this template, choose Tools | Templates
003 * and open the template in the editor.
004 */
005 package snmplogic.tareas;
006
007 import java.lang.Object;
008 import java.util.ArrayList;
009 import java.util.Arrays;
010 import java.util.List;
011 import java.util.regex.Matcher;
012 import java.util.regex.Pattern;
013 import snmplogic.almacenamiento.Data;
014 import snmplogic.ShowRunParser;
015
016 /**
017 *
018 * @author RAM
019 */
020 public class ShowRunInfo {
021
022     ShowRunParser showRunParser = new ShowRunParser();
023     Data data = new Data();
024
025     public ShowRunInfo() {
026         data = data.getInstance();
027     }
028
029     public boolean passwordHabilitado() {
030         return showRunParser.findOneLineWithPattern("enable password
<StringOb>");
031     }
032
033     public boolean secretHabilitado() {
034         return showRunParser.findLineStartsWith("enable secret 5");
035     }
036
037     public String[] lines() {
038         String[] lineasDeInteres =
showRunParser.getAllLinesReferenceAndPattern("line <StringOb> <intOb> <intOp>",
" password <StringOb>");
039         List<String> listaTemp = new ArrayList<String>();
040         for (int i = 0; i < lineasDeInteres.length; i++) {
```

```

041         if (lineasDeInteres[i].contains("line") &&
!lineasDeInteres[i].contains("password")) {
042
043             if (lineasDeInteres[i].length() == 10) {
044
045                 listaTemp.add(lineasDeInteres[i].substring(5, 10));
046                 if (i + 1 < lineasDeInteres.length && lineasDeInteres[i +
1].contains("password")) {
047                     int finPassword = lineasDeInteres[i + 1].length();
048
049                     listaTemp.add(lineasDeInteres[i + 1].substring(10, finPassword));
050                 } else {
051
052                     listaTemp.add("none");
053                 }
054             } else {
055                 int firstLine = Integer.parseInt(lineasDeInteres[i].substring(9, 10));
056                 int lastLine = Integer.parseInt(lineasDeInteres[i].substring(11, 12));
057                 for (int j = firstLine; j <= lastLine; j++) {
058
059                     listaTemp.add(lineasDeInteres[i].substring(5, 8) + " " + j);
060                     if (i + 1 < lineasDeInteres.length && lineasDeInteres[i +
1].contains("password")) {
061                         int finPassword = lineasDeInteres[i + 1].length();
062
063                         listaTemp.add(lineasDeInteres[i + 1].substring(10, finPassword));
064                     } else {
065
066                         listaTemp.add("ninguna");
067                     }
068                 }
069             }
070
071         }
072     }
073
074 }
075
076 String[] retorno = new String[listaTemp.size() / 2];
077 boolean switcher = true;
078 for (int k = 0; k < listaTemp.size(); k++) {
079     if (switcher) {
080         retorno[k / 2] = listaTemp.get(k);
081         switcher = !switcher;
082     } else {
083
084         switcher = !switcher;
085     }
086 }
087 return retorno;

```

```

088
089 }
090
091 public Object[][] lineInfo() {
092     String[] lineasDeInteres =
showRunParser.getAllLinesReferenceAndPattern("line <StringOb> <intOb> <intOp>",
" password <StringOb>");
093     List<String> listaTemp = new ArrayList<String>();
094     for (int i = 0; i < lineasDeInteres.length; i++) {
095         if (lineasDeInteres[i].contains("line") &&
!lineasDeInteres[i].contains("password")) {
096
097             if (lineasDeInteres[i].length() == 10) {
098
099                 listaTemp.add(lineasDeInteres[i].substring(5, 10));
100                 if (i + 1 < lineasDeInteres.length && lineasDeInteres[i +
1].contains("password")) {
101                     int finPassword = lineasDeInteres[i + 1].length();
102
103                     listaTemp.add(lineasDeInteres[i + 1].substring(10, finPassword));
104                 } else {
105
106                     listaTemp.add("none");
107                 }
108             } else {
109                 int firstLine = Integer.parseInt(lineasDeInteres[i].substring(9, 10));
110                 int lastLine = Integer.parseInt(lineasDeInteres[i].substring(11, 12));
111                 for (int j = firstLine; j <= lastLine; j++) {
112
113                     listaTemp.add(lineasDeInteres[i].substring(5, 8) + " " + j);
114                     if (i + 1 < lineasDeInteres.length && lineasDeInteres[i +
1].contains("password")) {
115                         int finPassword = lineasDeInteres[i + 1].length();
116
117                         listaTemp.add(lineasDeInteres[i + 1].substring(10, finPassword));
118                     } else {
119
120                         listaTemp.add("none");
121                     }
122                 }
123             }
124
125         }
126     }
127
128 }
129
130 Object[][] retorno = new Object[listaTemp.size() / 2][2];
131 boolean switcher = true;
132 for (int k = 0; k < listaTemp.size(); k++) {

```

```

133     if (switcher) {
134         retorno[k / 2][0] = listaTemp.get(k);
135         switcher = !switcher;
136     } else {
137         retorno[k / 2][1] = listaTemp.get(k);
138         switcher = !switcher;
139     }
140 }
141 return retorno;
142
143 }
144
145 public Object[][] userInfo() {
146     String[] lineasUsuarios = showRunParser.findAllLinesStartWith("username");
147     int lineasUsuarioLength = lineasUsuarios.length;
148     String[][] retorno = new String[lineasUsuarioLength][3];
149     for (int i = 0; i < lineasUsuarioLength; i++) {
150         String[] palabrasEnLinea = lineasUsuarios[i].split(" ");
151         int palabrasEnLineaLength = palabrasEnLinea.length;
152
153         boolean userFound = false;
154         boolean privilegeFound = false;
155         //boolean secretOrPasswordFound = false;
156         for (int j = 0; j < palabrasEnLineaLength; j++) {
157
158
159             if (palabrasEnLinea[j].equals("username") && !userFound) {
160                 j++;
161                 retorno[i][0] = palabrasEnLinea[j];
162                 userFound = true;
163             } else if (palabrasEnLinea[j].equals("privilege") && !privilegeFound) {
164                 j++;
165                 retorno[i][1] = palabrasEnLinea[j];
166                 privilegeFound = true;
167             } else if (palabrasEnLinea[j].equals("secret") ||
palabrasEnLinea[j].equals("password")) {
168                 retorno[i][2] = palabrasEnLinea[j];
169                 j = palabrasEnLineaLength;
170                 //secretOrPasswordFound = true;
171             }
172
173         }
174         if (!privilegeFound) {
175             retorno[i][1] = "default";
176         }
177     }
178 }
179 return retorno;
180 }
181

```

```

182 public Object[][] ntpInfo() {
183     String[] lineasServidoresNTP = showRunParser.findAllLinesStartWith("ntp
server");
184     int lineasServidoreNTPLength = lineasServidoresNTP.length;
185     String[][] retorno = new String[lineasServidoreNTPLength][3];
186     boolean prefer = false;
187
188     for (int i = 0; i < lineasServidoreNTPLength; i++) {
189         prefer = lineasServidoresNTP[i].contains("prefer");
190         String[] palabrasEnLinea = lineasServidoresNTP[i].split(" ");
191         //int palabrasEnLineaLength = palabrasEnLinea.length;
192         retorno[i][0] = palabrasEnLinea[0];
193         retorno[i][1] = palabrasEnLinea[2];
194         if (prefer) {
195             retorno[i][2] = "Si";
196         } else {
197             retorno[i][2] = "No";
198         }
199     }
200     return retorno;
201 }
202
203 public Object[][] conexionInfo() {
204     String[] lineasDeInteres =
showRunParser.getAllLinesReferenceAndPattern("line <StringOb> <intOb> <intOp>",
" exec-timeout <largeintOb> <largeintOb>");
205     List<String> listaTemp = new ArrayList<String>();
206
207     for (int i = 0; i < lineasDeInteres.length; i++) {
208         if (lineasDeInteres[i].contains("line")) {
209             String[] strLines = showRunParser.getDataFromLine(lineasDeInteres[i],
"line <StringOb> <intOb> <intOp>");
210             int intInicio = 0;
211             int intFin = 0;
212             if (strLines.length == 2) {
213                 intInicio = Integer.parseInt(strLines[1]);
214                 intFin = Integer.parseInt(strLines[1]);
215             } else if (strLines.length == 3) {
216                 intInicio = Integer.parseInt(strLines[1]);
217                 intFin = Integer.parseInt(strLines[2]);
218             }
219             for (int j = intInicio; j <= intFin; j++) {
220
221                 listaTemp.add(strLines[0] + " " + j);
222                 if (i + 1 < lineasDeInteres.length && lineasDeInteres[i +
1].contains("exec-timeout")) {
223                     i++;
224                     String[] strlineData =
showRunParser.getDataFromLine(lineasDeInteres[i], " exec-timeout <largeintOb>
<largeintOb>");

```

```

225         for (int k = 0; k < strlineData.length; k++) {
226             listaTemp.add(strlineData[k]);
227         }
228
229     } else {
230
231         listaTemp.add("default min");
232         listaTemp.add("default sec");
233     }
234 }
235 }
236
237 }
238
239 Object[][] retorno = new Object[listaTemp.size() / 3][3];
240
241 for (int k = 0; k < listaTemp.size(); k++) {
242
243     if ((k % 3) == 0) {
244         retorno[k / 3][0] = listaTemp.get(k);
245     } else if ((k % 3) == 1) {
246         retorno[k / 3][1] = listaTemp.get(k);
247     } else {
248         retorno[k / 3][2] = listaTemp.get(k);
249     }
250 }
251 return retorno;
252
253 }
254
255 public String intentosDeConexion() {
256     String retorno = null;
257     String[] temp = showRunParser.getDataFirstLineInText("security
authentication failure rate <largeintOb> log");
258     if (temp == null) {
259         String[] strDefault = {"Deafault"};
260         return strDefault[0];
261     } else {
262         retorno = temp[0];
263     }
264
265     return retorno;
266 }
267
268 public String[][] nivelesInfo() {
269     String[] lineasDeInteres = showRunParser.findAllLinesStartWith("enable
secret");
270     int intLineasDeInteresLength = lineasDeInteres.length;
271     String[][] retorno = new String[15][2];
272     for (int i = 1; i <= 15; i++) {

```

```

273     retorno[i - 1][0] = "level " + i;
274     retorno[i - 1][1] = "deshabilitado";
275     for (int j = 0; j < intLineasDeInteresLength; j++) {
276         if (lineasDeInteres[j].contains("level")) {
277             if (lineasDeInteres[j].contains("level " + i)) {
278                 retorno[i - 1][1] = "habilitado";
279             }
280         } else {
281             retorno[14][1] = "habilitado";
282         }
283     }
284 }
285 return retorno;
286 }
287
288 public String[] vistasList() {
289     String[] lineasDeInteres = showRunParser.findAllLinesStartWith("snmp-server
view");
290     Arrays.sort(lineasDeInteres);
291     List<String> listaNoRepetida = new ArrayList<String>();
292     String[] vistas = new String[lineasDeInteres.length];
293     for (int i = 0; i < lineasDeInteres.length; i++) {
294         String[] palabrasEnLinea = lineasDeInteres[i].split(" ");
295
296         vistas[i] = palabrasEnLinea[2];
297         if (i == 0) {
298             listaNoRepetida.add(vistas[i]);
299         } else {
300             if (!vistas[i].equals(vistas[i - 1])) {
301                 listaNoRepetida.add(vistas[i]);
302             }
303         }
304     }
305     String[] retorno = listaNoRepetida.toArray(new String[0]);
306     return retorno;
307 }
308
309 public String[][] vistasDetailedInfo() {
310     //Lineas de interes
311     String[] lineasDeInteres = showRunParser.findAllLinesStartWith("snmp-server
view");
312     Arrays.sort(lineasDeInteres, String.CASE_INSENSITIVE_ORDER);
313     int lineasDeInteresLength = lineasDeInteres.length;
314     String[][] arregloCondensado = new String[lineasDeInteresLength][3];
315     //Obtenemos lo importante
316     for (int i = 0; i < lineasDeInteresLength; i++) {
317         String[] palabrasEnLinea = lineasDeInteres[i].split(" ");
318         //Nombre
319         arregloCondensado[i][0] = palabrasEnLinea[2];
320         //MIB

```

```

321     arregloCondensado[i][1] = palabrasEnLinea[3];
322     //Included Excluded
323     arregloCondensado[i][2] = palabrasEnLinea[4];
324 }
325 return arregloCondensado;
326 }
327
328 public String[] aclList() {
329     String[] lineasDeInteres = showRunParser.findAllLinesStartWith("access-
list");
330     Arrays.sort(lineasDeInteres);
331     List<String> listaNoRepetida = new ArrayList<String>();
332     String[] vistas = new String[lineasDeInteres.length];
333     for (int i = 0; i < lineasDeInteres.length; i++) {
334         String[] palabrasEnLinea = lineasDeInteres[i].split(" ");
335
336         vistas[i] = palabrasEnLinea[1];
337         if (i == 0) {
338             listaNoRepetida.add(vistas[i]);
339         } else {
340             if (!vistas[i].equals(vistas[i - 1])) {
341                 listaNoRepetida.add(vistas[i]);
342             }
343         }
344     }
345     String[] retorno = listaNoRepetida.toArray(new String[0]);
346     return retorno;
347 }
348
349 public String[][] accessListDetailedInfo() {
350     //Lineas de interes
351     String[] lineasDeInteres = showRunParser.findAllLinesStartWith("access-
list");
352     Arrays.sort(lineasDeInteres, String.CASE_INSENSITIVE_ORDER);
353     int lineasDeInteresLength = lineasDeInteres.length;
354     String[][] arregloCondensado = new String[lineasDeInteresLength][3];
355     //Obtenemos lo importante
356     for (int i = 0; i < lineasDeInteresLength; i++) {
357         String[] palabrasEnLinea = lineasDeInteres[i].split(" ");
358         //Numero
359         arregloCondensado[i][0] = palabrasEnLinea[1];
360         //permit deny
361         arregloCondensado[i][1] = palabrasEnLinea[2];
362         if (palabrasEnLinea[2].equals("deny")) {
363             //Host
364             arregloCondensado[i][2] = palabrasEnLinea[5];
365             if (palabrasEnLinea[5].equals("any") && palabrasEnLinea.length > 6) {
366                 arregloCondensado[i][2] = arregloCondensado[i][2] + " " +
palabrasEnLinea[6];
367             }

```

```

368     } else {
369         //Host
370         arregloCondensado[i][2] = palabrasEnLinea[3];
371
372     }
373 }
374 return arregloCondensado;
375 }
376
377 public String[][] anmpServerCommunityInfo() {
378     //Lineas de interes
379     String[] lineasDeInteres = showRunParser.findAllLinesStartWith("snmp-server
community");
380     Arrays.sort(lineasDeInteres, String.CASE_INSENSITIVE_ORDER);
381     int lineasDeInteresLength = lineasDeInteres.length;
382     String[][] arregloCondensado = new String[lineasDeInteresLength][4];
383     //Obtenemos lo importante
384     for (int i = 0; i < lineasDeInteresLength; i++) {
385         String[] palabrasEnLinea = lineasDeInteres[i].split(" ");
386         //Nombre comunidad
387         arregloCondensado[i][0] = palabrasEnLinea[2];
388         if (palabrasEnLinea.length >= 5) {
389             //Vista
390             arregloCondensado[i][1] = palabrasEnLinea[4];
391         }
392         if (palabrasEnLinea.length >= 6) {
393             //Tipo
394             arregloCondensado[i][2] = palabrasEnLinea[5];
395         }
396         if (palabrasEnLinea.length >= 7) {
397             //ACL
398             arregloCondensado[i][3] = palabrasEnLinea[6];
399         }
400
401
402     }
403     return arregloCondensado;
404 }
405
406 public String[][] privilegiosInfo() {
407     String[] lineasDeInteres = showRunParser.findAllLinesStartWith("privilege
exec");
408     int intLineasDeInteresLength = lineasDeInteres.length;
409     String[][] retorno = new String[intLineasDeInteresLength][3];
410     for (int i = 0; i < intLineasDeInteresLength; i++) {
411         String linea = lineasDeInteres[i];
412         linea = linea.substring(15);
413         if (linea.startsWith("all")) {
414             retorno[i][2] = "all";
415             linea = linea.substring(4);

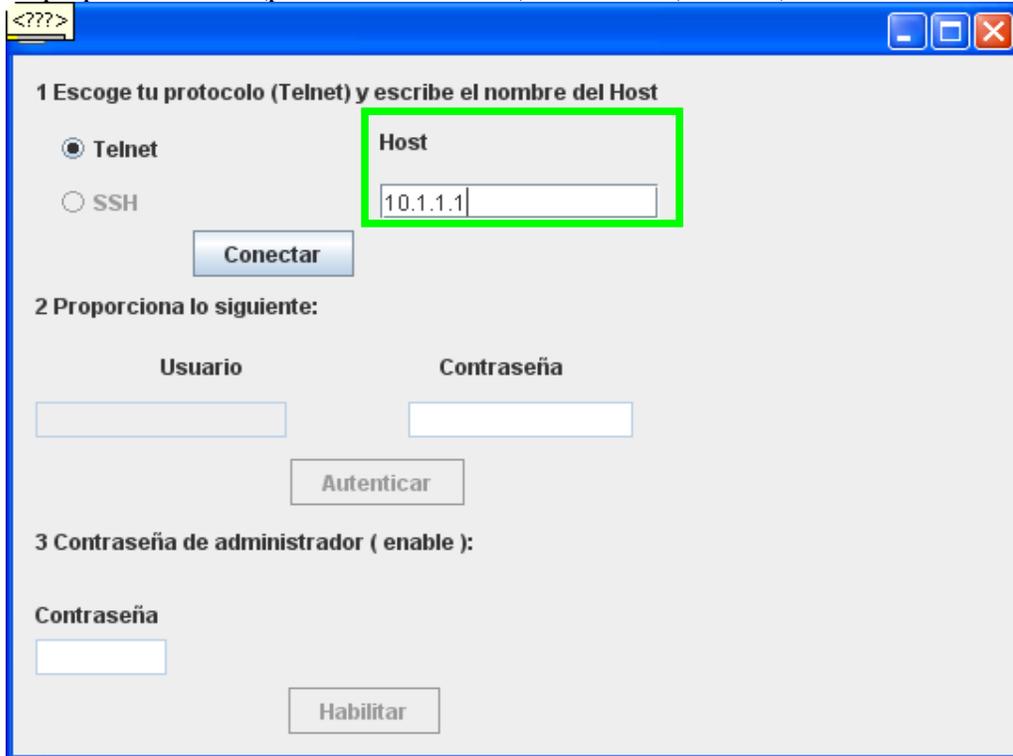
```

```
416     }
417     linea = linea.substring(6);
418     Pattern patron = Pattern.compile("[0-9]+");
419     Matcher comparador = patron.matcher(linea);
420     comparador.find();
421     retorno[i][1] = comparador.group();
422     linea = linea.substring(comparador.end() + 1);
423     retorno[i][0] = linea;
424
425
426     }
427     return retorno;
428
429 }
430 }
```

## Pruebas

Prueba 001: Conexión, autenticación y habilitación como administrador exitosa.  
Desarrollo

Se proporciona la IP (protocolo de Internet) del Router (ruteador).



The image shows a software window titled "<???" with standard Windows window controls. The window contains three sections for configuring a Telnet connection:

- 1 Escoge tu protocolo (Telnet) y escribe el nombre del Host:** This section has two radio buttons: "Telnet" (which is selected) and "SSH". To the right is a "Host" label above a text input field containing "10.1.1.1". A blue "Conectar" button is positioned below these options.
- 2 Proporciona lo siguiente:** This section has two labels, "Usuario" and "Contraseña", each above a text input field. A blue "Autenticar" button is centered below these fields.
- 3 Contraseña de administrador ( enable ):** This section has a "Contraseña" label above a text input field. A blue "Habilitar" button is centered below the field.

Se proporciona un nombre de usuario y una contraseña válidos ambos.

<???

1 Escoge tu protocolo (Telnet) y escribe el nombre del Host

Telnet      Host

SSH      10.1.1.1

Conectar

2 Proporciona lo siguiente:

Usuario      Contraseña

hector      .....

Autenticar

3 Contraseña de administrador ( enable ):

Contraseña

.....

Habilitar

Se proporciona una contraseña de administrador válida.

1 Escoge tu protocolo (Telnet) y escribe el nombre del Host

Telnet      Host

SSH      10.1.1.1

Conectar

2 Proporciona lo siguiente:

Usuario      Contraseña

hector      .....

Autenticar

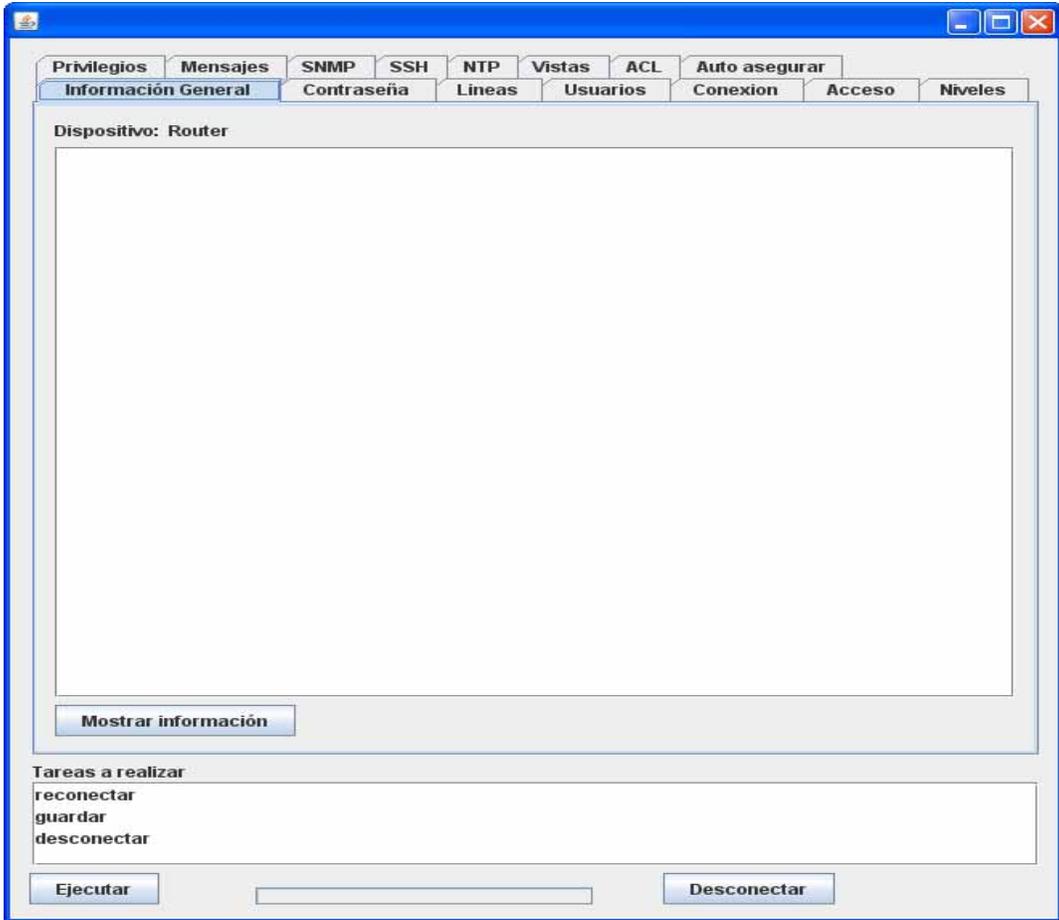
3 Contraseña de administrador ( enable ):

Contraseña

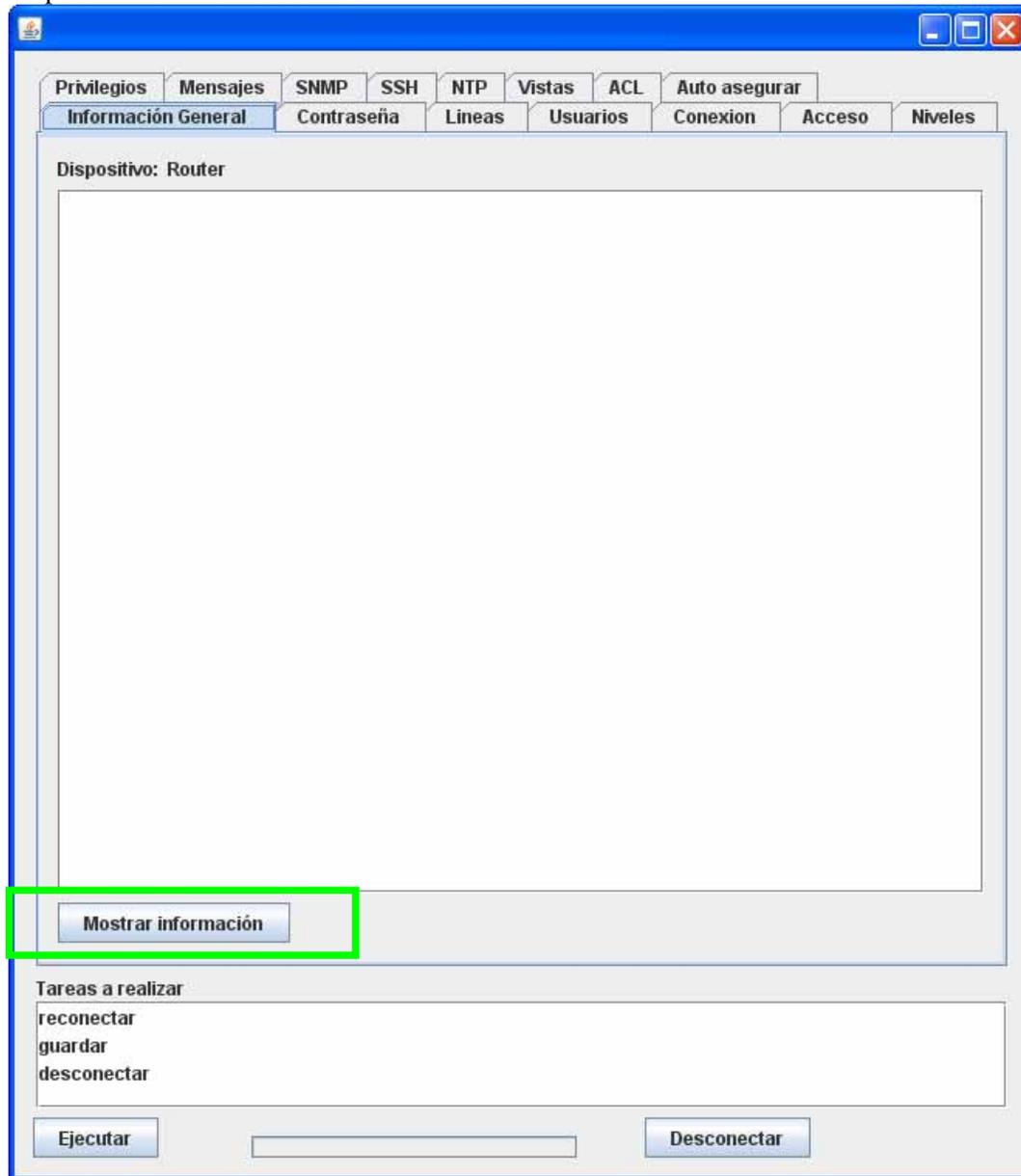
.....

Habilitar

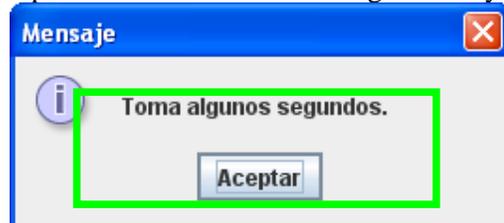
Se logra el acceso a la interfaz principal de configuración del Router (ruteador).



Prueba 002: Mostrar información del Router (ruteador).  
Se presiona el botón de demostrar información.



Aparece una ventana con la siguiente leyenda.



Se muestra la información del Router (ruteador).

The screenshot displays a web-based management interface for a Cisco router. The top navigation bar includes tabs for Privilegios, Mensajes, SNMP, SSH, NTP, Vistas, ACL, and Auto asegurar. Below this, a secondary bar contains tabs for Información General, Contraseña, Líneas, Usuarios, Conexión, Acceso, and Niveles. The main content area, titled 'Dispositivo: Router', shows the output of the 'show tech-support' command. The output includes the Cisco IOS version (12.4(18)), copyright information, ROM details, and system uptime (38 minutes). A green rectangular box highlights the entire output text. Below the text area is a 'Mostrar información' button. At the bottom, there is a 'Tareas a realizar' section with a list of actions: 'reconectar', 'guardar', and 'desconectar'. Below this list are 'Ejecutar' and 'Desconectar' buttons, with an empty input field between them.

Privilegios Mensajes SNMP SSH NTP Vistas ACL Auto asegurar

Información General Contraseña Líneas Usuarios Conexión Acceso Niveles

**Dispositivo: Router**

```
show tech-support
show tech-support

----- show version -----

Cisco IOS Software, C1700 Software (C1700-ADVSECURITYK9-M), Version 12.4(18), RELEASE SOFTWARE
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 1986-2007 by Cisco Systems, Inc.
Compiled Fri 30-Nov-07 13:45 by prod_rel_team

ROM: ROMMON Emulation Microcode
ROM: C1700 Software (C1700-ADVSECURITYK9-M), Version 12.4(18), RELEASE SOFTWARE (fc1)

Router uptime is 38 minutes
System returned to ROM by unknown reload cause - suspect boot_data[BOOT_COUNT] 0x0, BOOT_COUN
System image file is "tftp://255.255.255.255/unknown"

This product contains cryptographic features and is subject to United
States and local country laws governing import, export, transfer and
use. Delivery of Cisco cryptographic products does not imply
third-party authority to import, export, distribute or use encryption.
Importers, exporters, distributors and users are responsible for
compliance with U.S. and local country laws. By using this product you
agree to comply with applicable laws and regulations. If you are unable
to comply with U.S. and local laws, return this product immediately.
```

Mostrar información

**Tareas a realizar**

- reconectar
- guardar
- desconectar

Ejecutar  Desconectar

Confirmación de la misma información en el Router (ruteador).

```
10.1.1.1 - PuTTY

User Access Verification

Username: hector
Password:

Router>enable
Password:
Router#show version
Cisco IOS Software, C1700 Software (C1700-ADVSECURITYK9-M), Version 12.4(18), RE
LEASE SOFTWARE (fc1)
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 1986-2007 by Cisco Systems, Inc.
Compiled Fri 30-Nov-07 13:45 by prod_rel_team

ROM: ROMMON Emulation Microcode
ROM: C1700 Software (C1700-ADVSECURITYK9-M), Version 12.4(18), RELEASE SOFTWARE
(fc1)

Router uptime is 39 minutes
System returned to ROM by unknown reload cause - suspect boot_data[BOOT_COUNT] 0
x0, BOOT_COUNT 0, BOOTDATA 19
System image file is "tftp://255.255.255.255/unknown"

This product contains cryptographic features and is subject to United
States and local country laws governing import, export, transfer and
use. Delivery of Cisco cryptographic products does not imply
third-party authority to import, export, distribute or use encryption.
Importers, exporters, distributors and users are responsible for
compliance with U.S. and local country laws. By using this product you
agree to comply with applicable laws and regulations. If you are unable
--More--
```

Prueba 003: Cambiar contraseña.

Punto de partida.

Privilegios Mensajes SNMP SSH NTP Vistas ACL Auto asegurar

Información General **Contraseña** Lineas Usuarios Conexion Acceso Niveles

Password **Habilitado**

Si deseas establecer un password nuevo:

Habilitar

**Deshabilitar**

-

**Añadir**

Tareas a realizar

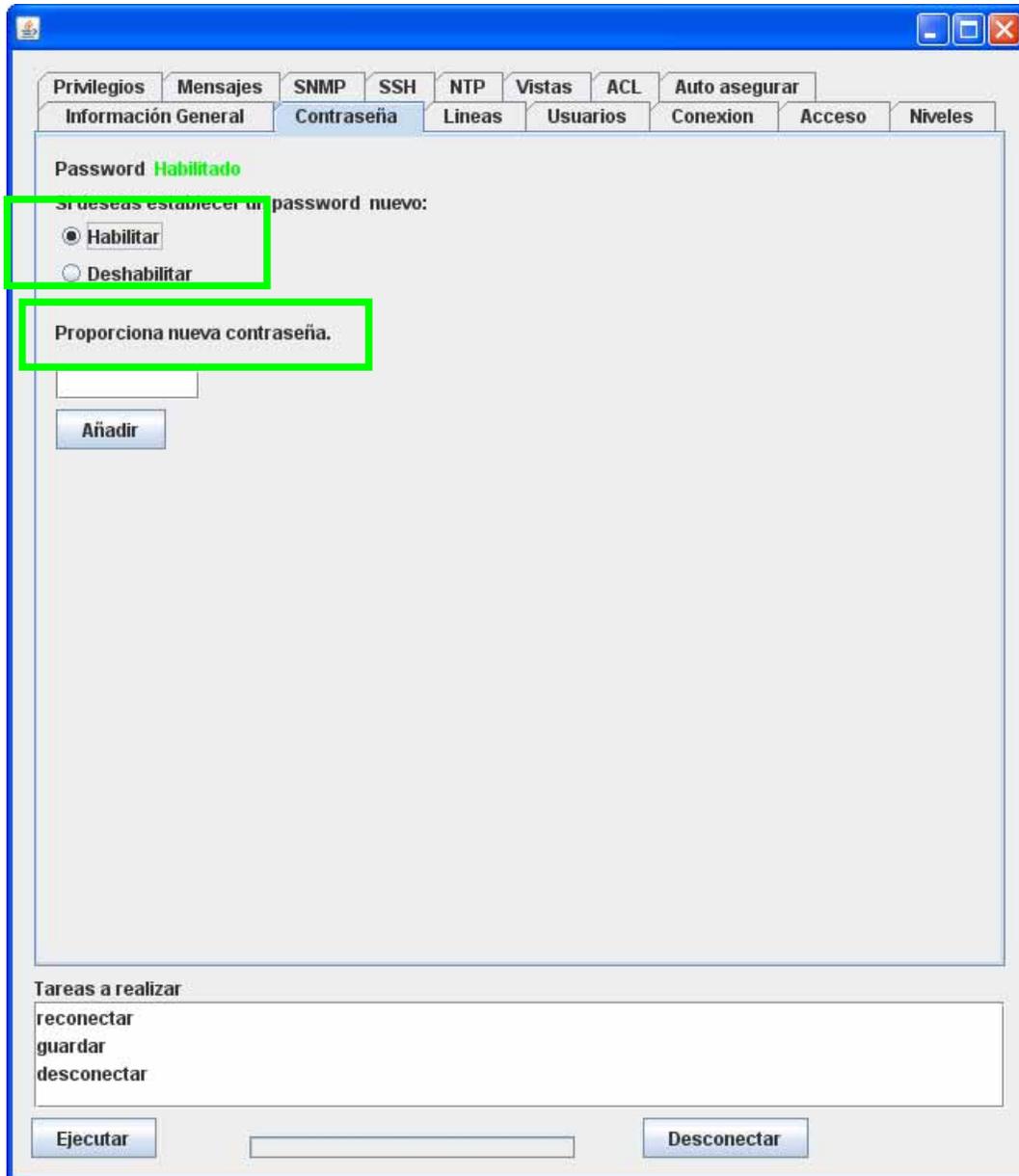
reconectar

guardar

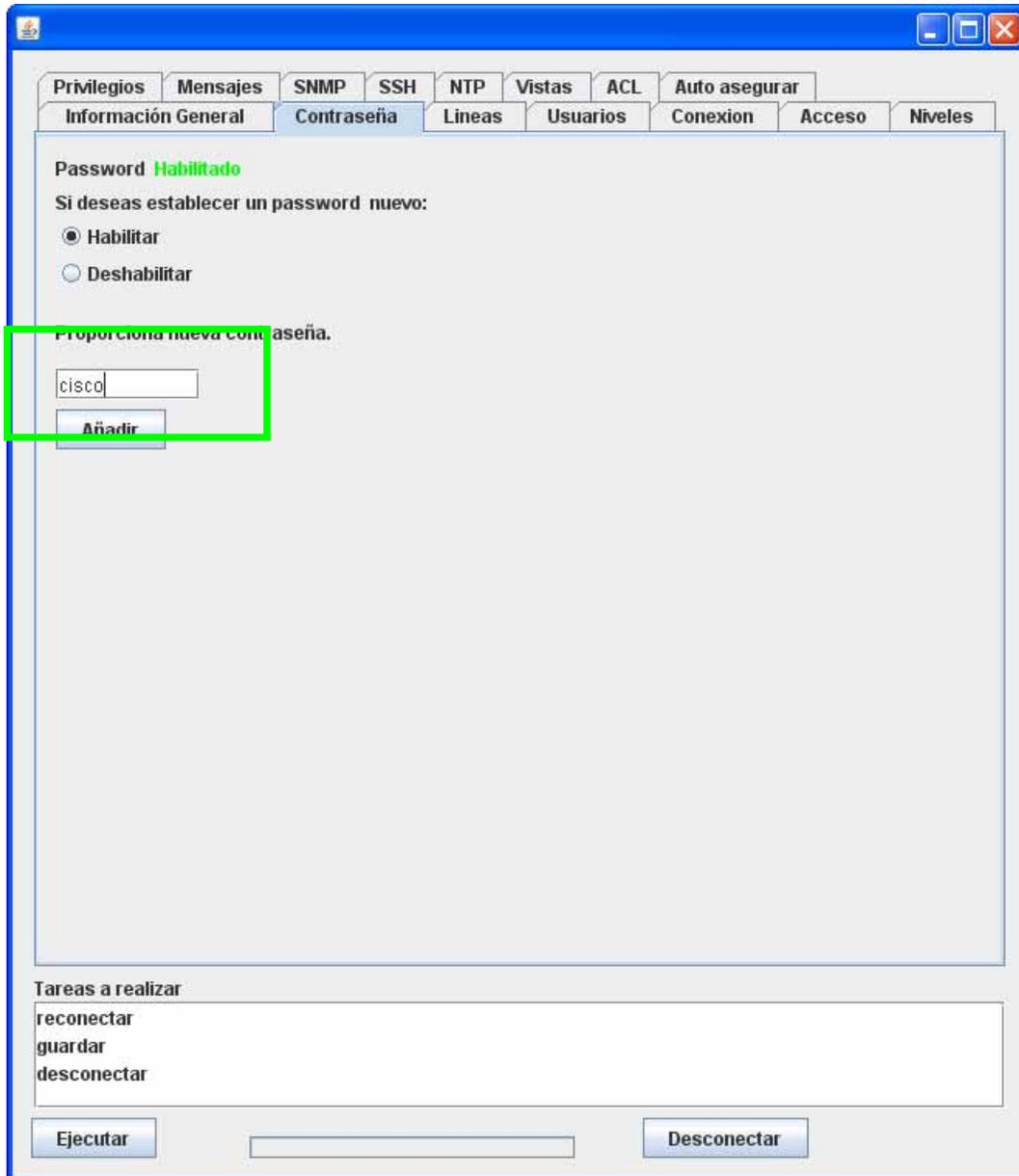
desconectar

**Ejecutar**  **Desconectar**

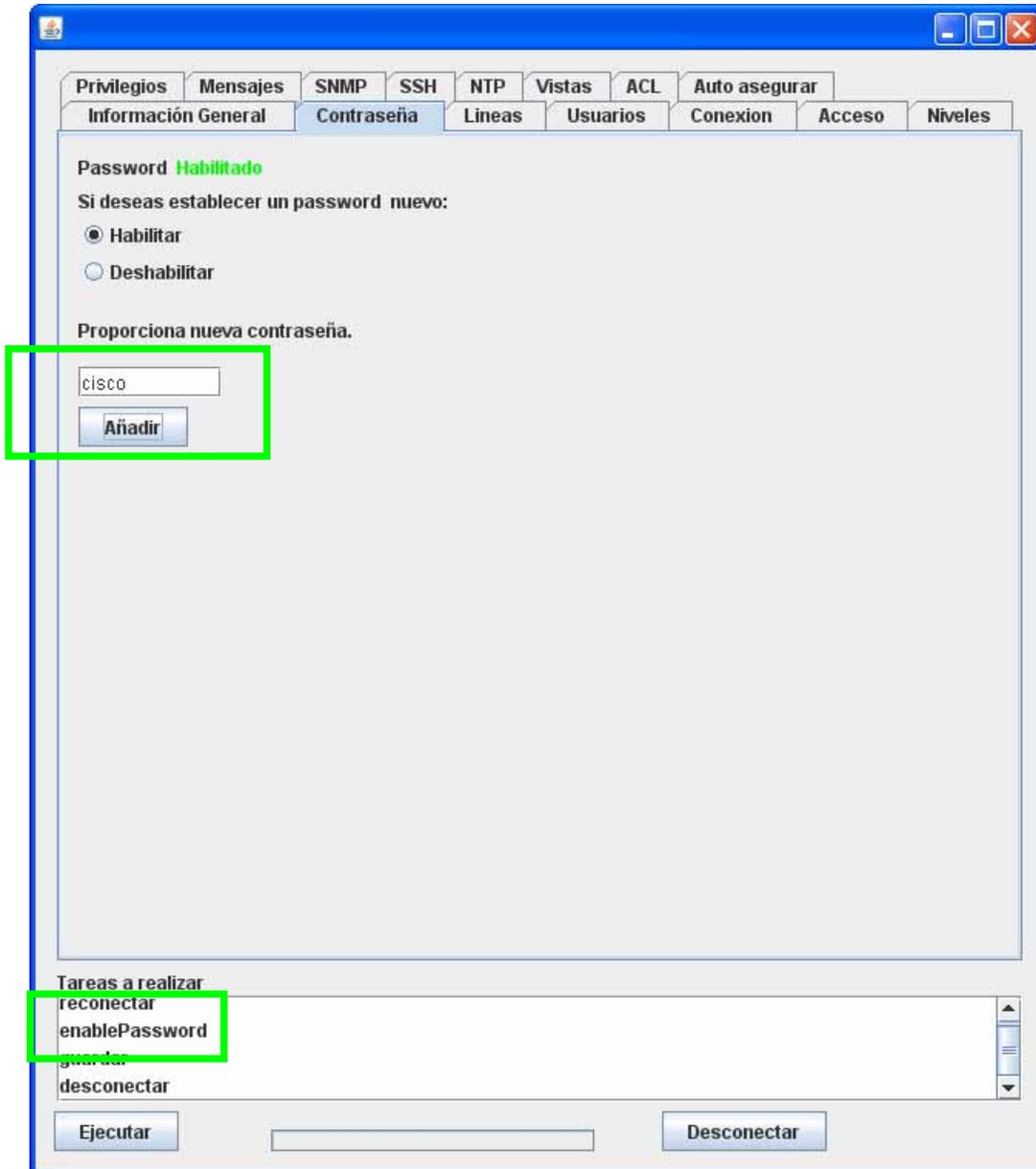
Se selecciona la opción de habilitar y la bandera de mensajes cambia.



Se proporciona la nueva contraseña.



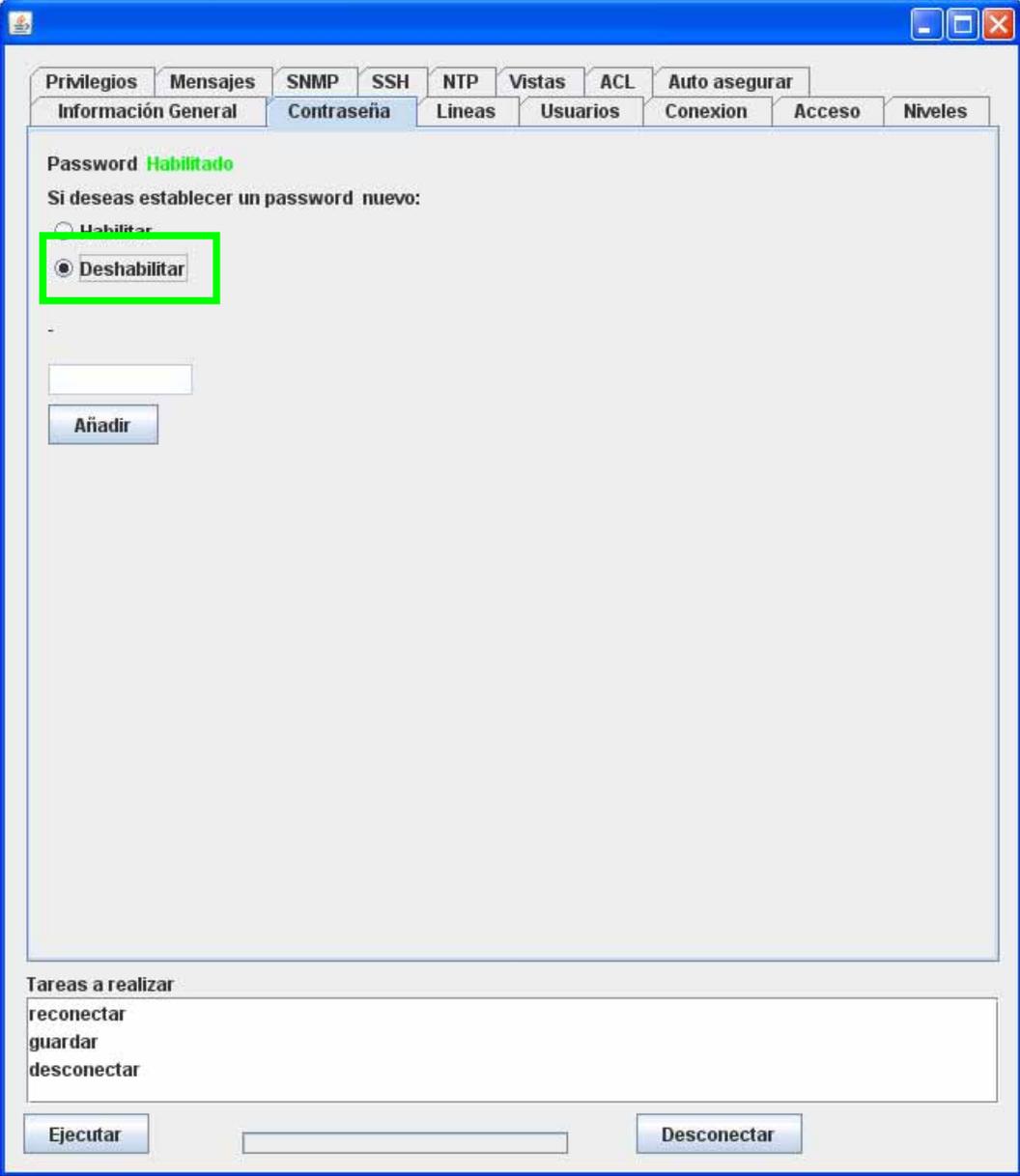
Se presiona el botón de añadir y la tarea se agrega a la lista de tareas ejecutar.



Comprobación del cambio en la configuración del Router (ruteador).

```
10.1.1.1 - PuTTY
Current configuration : 2174 bytes
!
version 12.4
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname Router
!
boot-start-marker
boot-end-marker
!
security authentication failure rate 2 log
enable secret level 5 5 $1$sGRI$ybtQz1Vy3ZubRP7pj/edU0
enable secret level 7 5 $1$d3Sf$7gOgzcp3/mlveVMct4K01
enable secret level 9 5 $1$jaK/$0nB1CJ9mFPu4zs6KbNzo91
enable secret level 10 5 $1$HVe9$Mcv9aUUUruxtFkw9UJq4e.
enable password cisco
!
aaa new-model
!
!
--More--
```

Prueba 004: Intento de deshabilitar contraseña.  
Se selecciona la opción de deshabilitar.



Privilegios Mensajes SNMP SSH NTP Vistas ACL Auto asegurar

Información General **Contraseña** Lineas Usuarios Conexion Acceso Niveles

Password **Habilitado**

Si deseas establecer un password nuevo:

Habilitar

**Deshabilitar**

-

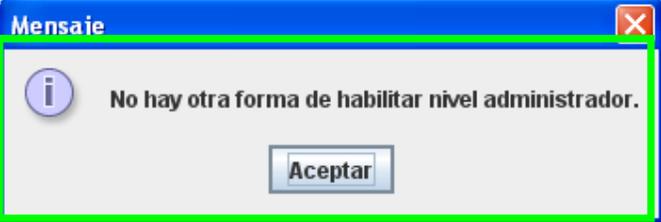
Añadir

Tareas a realizar

reconectar  
guardar  
desconectar

Ejecutar  Desconectar

Como no está establecido otro medio de autenticación para la conexión con el Router (ruteador), por ejemplo la contraseña secreta (secret). El Router (ruteador) manda siguiente mensaje de advertencia. Y regresa al paso anterior.



Mensaje

**i** No hay otra forma de habilitar nivel administrador.

Aceptar

Prueba 005: cambio de contraseña en líneas  
Punto de partida.

The screenshot shows a network configuration window with a blue border. At the top, there are several tabs: Privilegios, Mensajes, SNMP, SSH, NTP, Vistas, ACL, and Auto asegurar. Below these, there are sub-tabs: Información General, Contraseña, Líneas, Usuarios, Conexion, Acceso, and Niveles. The 'Líneas' sub-tab is selected. On the left side, there is a 'Linea' dropdown menu currently set to 'con 0', a 'Nueva contraseña' text input field, and an 'Añadir' button. The main area contains a table with two columns: 'Linea' and 'Contraseña'. The table lists several lines and their passwords. The row for 'vty 1' with password 'mrea' is highlighted with a green border. Below the table, there is a 'Tareas a realizar' section with a list of actions: reconectar, guardar, and desconectar. At the bottom, there are 'Ejecutar' and 'Desconectar' buttons, with an empty text input field between them.

| Linea | Contraseña |
|-------|------------|
| con 0 | none       |
| aux 0 | none       |
| vty 0 | linea      |
| vty 1 | mrea       |
| vty 2 | jijiji     |
| vty 3 | hola       |
| vty 4 | adios      |

Comprobación de la misma información dentro del Router (ruteador).

```
10.1.1.1 - PuTTY
line vty 0
  exec-timeout 1 1
  password linea
line vty 1
  password linea
line vty 2
  exec-timeout 3 2
  password jijiji
line vty 3
  password hola
line vty 4
  password adios
parser view vista1
secret 5 $1$TVOB$E5HmQ3ZnI4zGeF1S2ch2C.
commands exec include all ping
commands exec include show running-config
commands exec include show
commands exec exclude debug
!
!
ntp server 10.1.1.3 prefer
ntp server 10.1.1.2 prefer
end
--More--
```

Se selecciona la línea que se desea modificar.

Privilegios Mensajes SNMP SSH NTP Vistas ACL Auto asegurar

Información General Contraseña Lineas Usuarios Conexion Acceso Niveles

Linea

vtty 2

Nueva contraseña

Añadir

| Linea  | Contraseña |
|--------|------------|
| con 0  | none       |
| aux 0  | none       |
| vtty 0 | linea      |
| vtty 1 | linea      |
| vtty 2 | jijiji     |
| vtty 3 | hola       |
| vtty 4 | adios      |

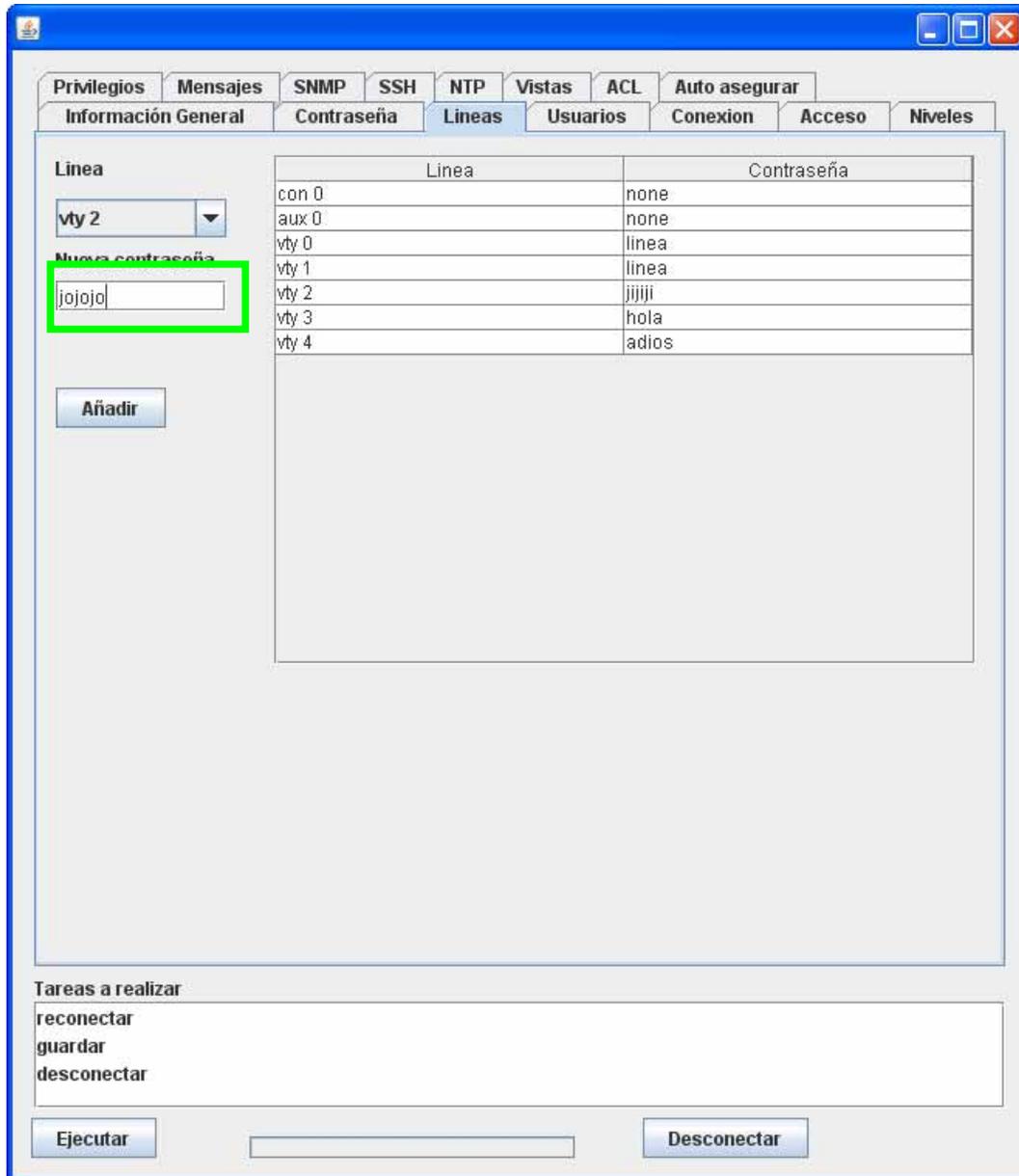
Tareas a realizar

reconectar  
guardar  
desconectar

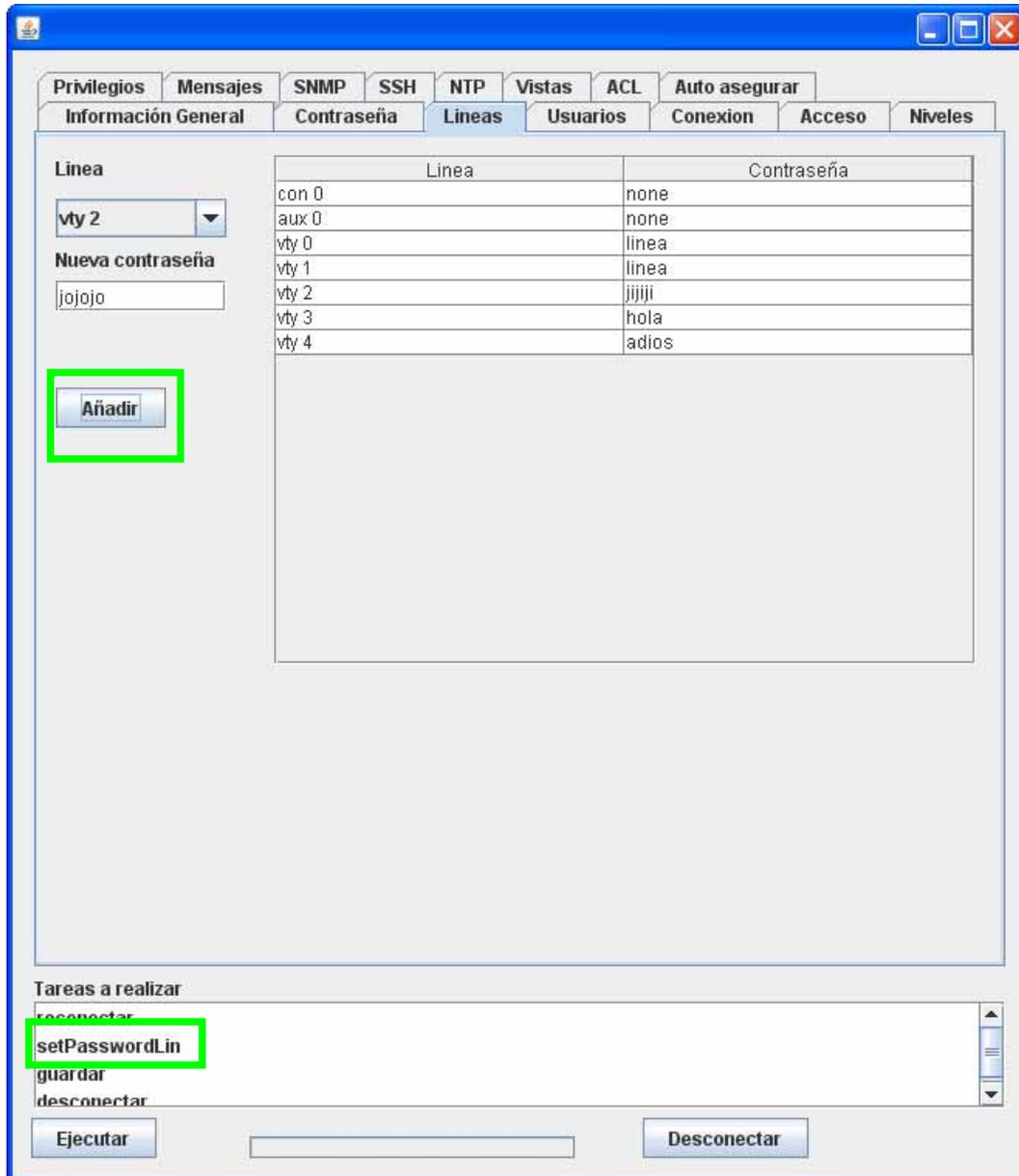
Ejecutar

Desconectar

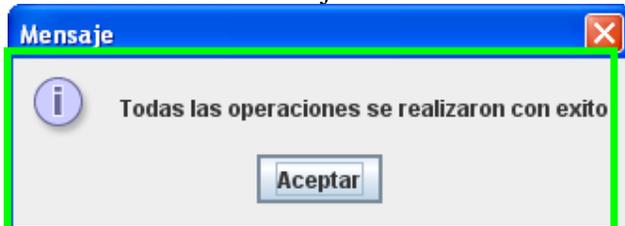
Se proporciona la nueva contraseña.



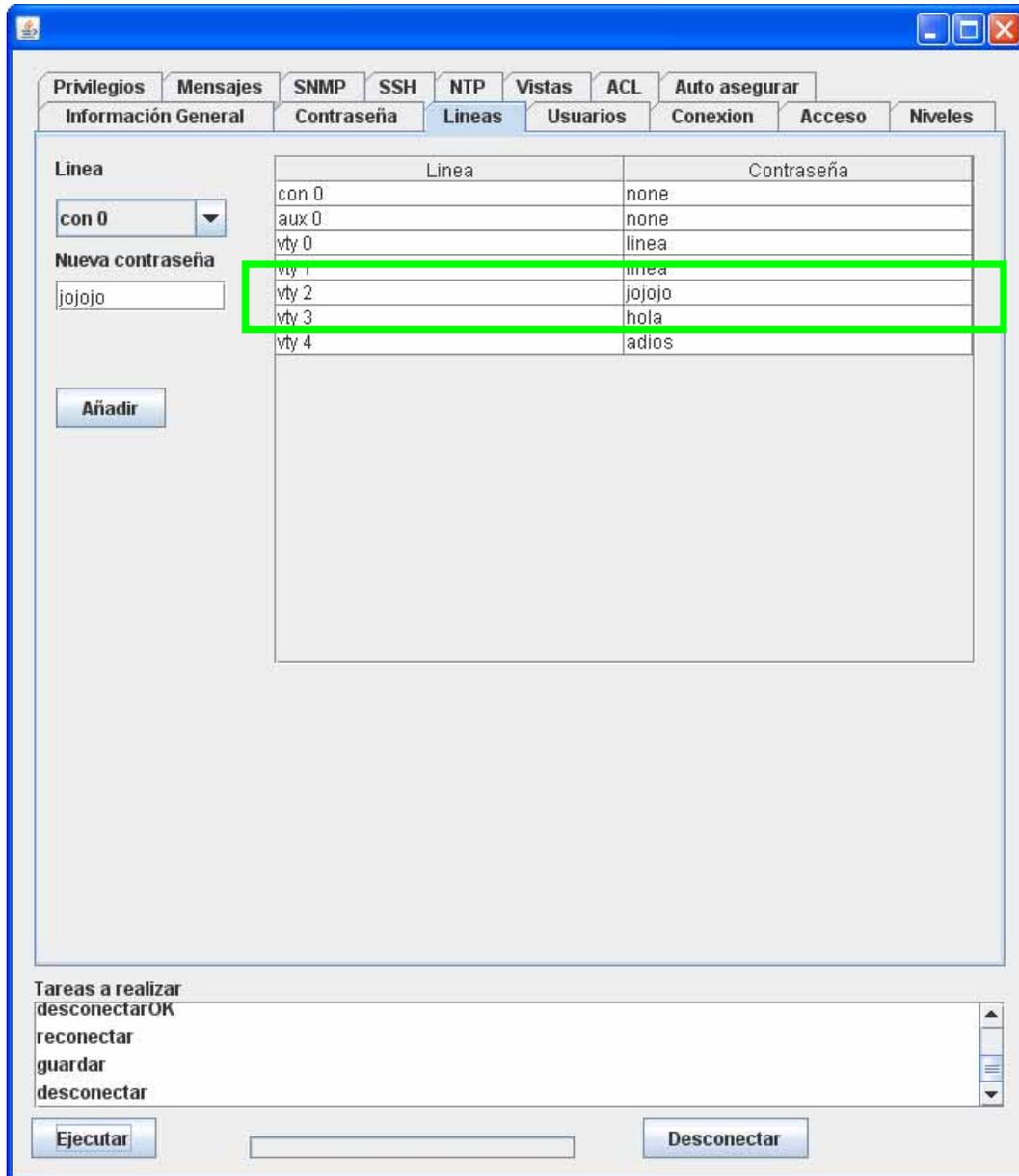
Se presiona el botón de añadir y se agrega automáticamente la tarea a la lista de tareas por ejecutar.



Presionamos el botón de ejecutar.



Se observan los cambios en la interfaz.



Se comprueban los cambios en la configuración del Router (ruteador).

```
10.1.1.1 - PuTTY
line vty 0
  exec-timeout 1 1
  password linea
line vty 1
  password linea
line vty 2
  exec-timeout 3 2
  password jojojo
line vty 3
  password hola
line vty 4
  password adios
parser view vista1
secret 5 $1$TVOB$E5HmQ3ZnI4zGeF1S2ch2C.
commands exec include all ping
commands exec include show running-config
commands exec include show
commands exec exclude debug
!
!
ntp server 10.1.1.3 prefer
ntp server 10.1.1.2 prefer
end
--More--
```

Prueba 006: agregar un nuevo usuario.  
Punto de partida.

The screenshot shows a configuration window with the following elements:

- Navigation tabs: Privilegios, Mensajes, SNMP, SSH, NTP, Vistas, ACL, Auto asegurar.
- Sub-tabs: Información General, Contraseña, Lineas, **Usuarios**, Conexion, Acceso, Niveles.
- Table of users:

| Usuario | Nivel | Encriptación |
|---------|-------|--------------|
| hector  | 15    | secret       |
| ramiro  | 15    | password     |
| pepe    | 6     | secret       |
| mm      | 12    | password     |

Below the table, there is an 'Eliminar' button and a form for adding a new user:

- Usuario:** [Empty text field]
- Encriptación:** [password] (dropdown menu)
- Contraseña:** [Empty text field]
- Nivel de privilegios:** 1 (slider bar)

At the bottom of the form is an 'Añadir' button. Below the form is a 'Tareas a realizar' section with a list: reconectar, guardar, desconectar. At the very bottom are 'Ejecutar' and 'Desconectar' buttons.

Se confirma la misma información dentro del Router (ruteador).



Privilegios Mensajes SNMP SSH NTP Vistas ACL Auto asegurar

Información General Contraseña Líneas Usuarios Conexion Acceso Niveles

| Usuario | Nivel | Encriptación |
|---------|-------|--------------|
| hector  | 15    | secret       |
| ramiro  | 15    | password     |
| pepe    | 6     | secret       |
| mm      | 12    | password     |

Eliminar

Usuario:  Encriptación:  Contraseña:

Nivel de privilegios: 1

Añadir

Tareas a realizar

reconectar  
guardar  
desconectar

Ejecutar  Desconectar

Se selecciona un tipo de encriptación y una contraseña.

Privilegios Mensajes SNMP SSH NTP Vistas ACL Auto asegurar

Información General Contraseña Líneas Usuarios Conexion Acceso Niveles

| Usuario | Nivel | Encriptación |
|---------|-------|--------------|
| hector  | 15    | secret       |
| ramiro  | 15    | password     |
| pepe    | 6     | secret       |
| mm      | 12    | password     |

Eliminar

Usuario: arturo

Encriptación: password

Contraseña: zuniga

Nivel de privilegios: 1

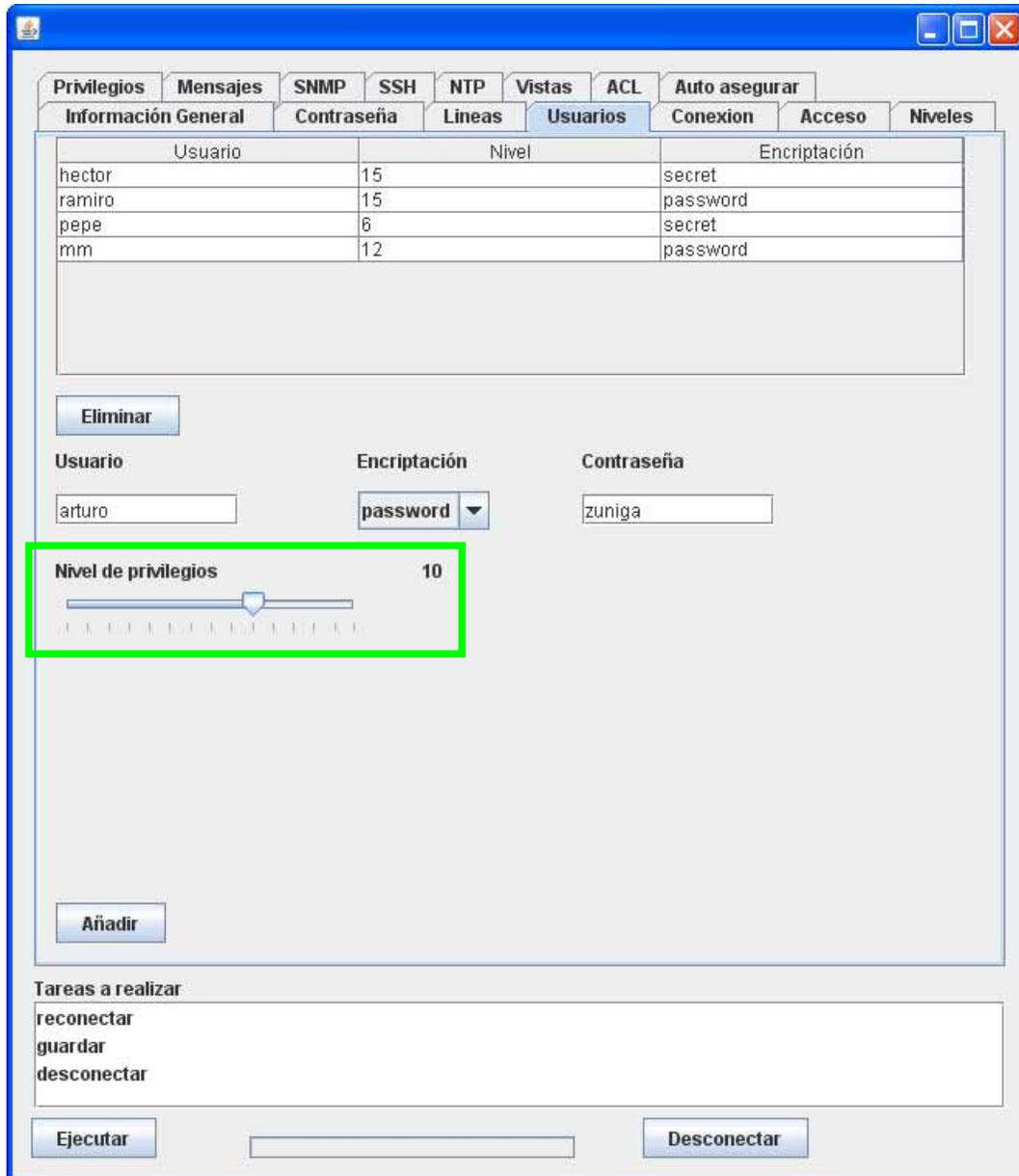
Añadir

Tareas a realizar:

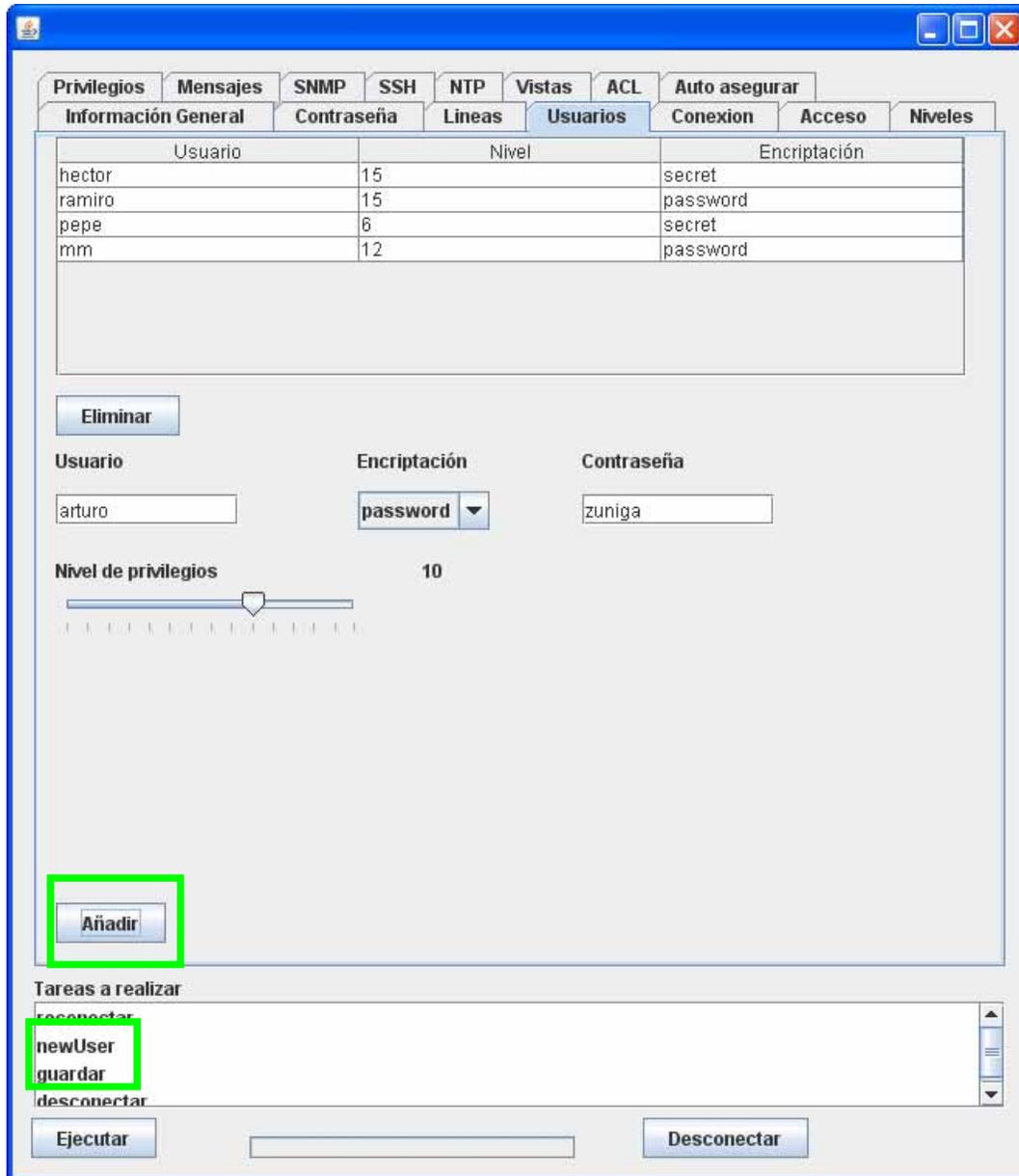
- reconectar
- guardar
- desconectar

Ejecutar Desconectar

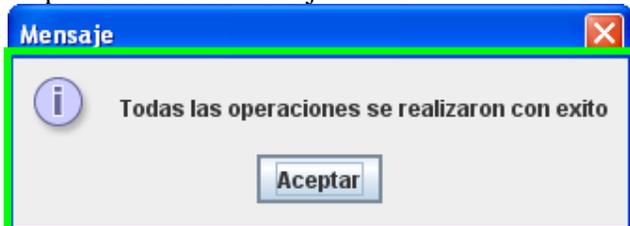
Se establece un nivel de privilegios para ese usuario.



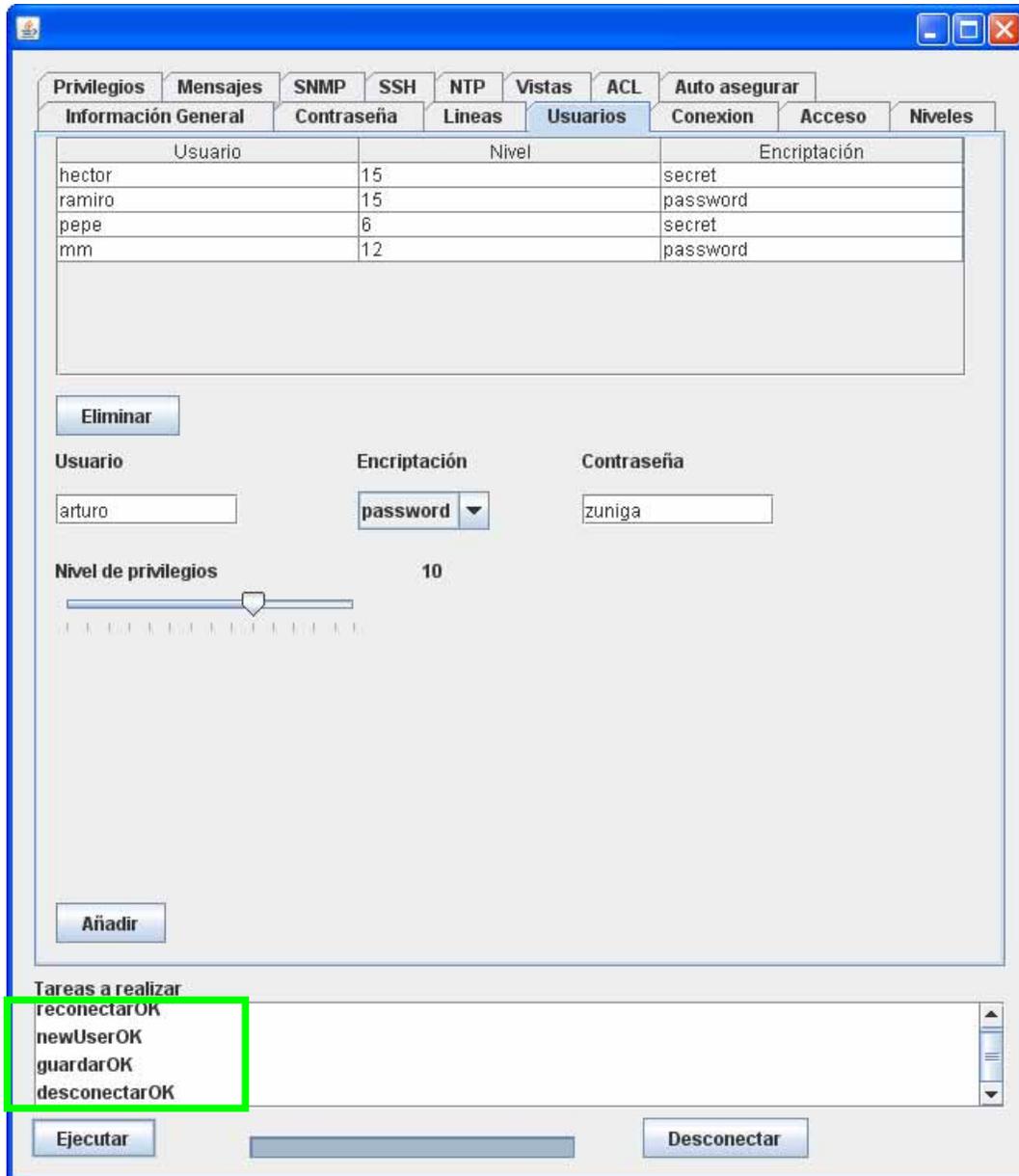
Se presiona el botón de añadir y automáticamente se agrega la tarea la lista de tareas ejecutar.



Se presiona el botón de ejecutar.



Se aprecia como todas las tareas tienen la etiqueta de que fueron realizadas correctamente.



Se ven los cambios en la interfaz.

The screenshot shows a web-based configuration interface for a router, specifically the 'Usuarios' (Users) tab. The interface includes a menu bar at the top with options like 'Privilegios', 'Mensajes', 'SNMP', 'SSH', 'NTP', 'Vistas', 'ACL', and 'Auto asegurar'. Below the menu, there are sub-tabs: 'Información General', 'Contraseña', 'Lineas', 'Usuarios', 'Conexion', 'Acceso', and 'Niveles'. The 'Usuarios' sub-tab is active, displaying a table of users.

| Usuario | Nivel | Encriptación |
|---------|-------|--------------|
| hector  | 15    | secret       |
| ramiro  | 15    | password     |
| pepe    | 6     | secret       |
| min     | 12    | password     |
| arturo  | 10    | password     |

Below the table, there is an 'Eliminar' button. Underneath, the configuration for the selected user 'arturo' is shown. It includes a text field for the username 'arturo', a dropdown menu for 'Encriptación' set to 'password', and a text field for the password 'zuniga'. A 'Nivel de privilegios' slider is set to 10. At the bottom left, there is an 'Añadir' button. The bottom of the interface features a 'Tareas a realizar' section with a list of actions: 'desconectarOK', 'reconectar', 'guardar', and 'desconectar'. At the very bottom, there are 'Ejecutar' and 'Desconectar' buttons.

Se comprueban los datos dentro de la configuración del Router (ruteador).



Prueba 007: eliminar a un usuario  
Punto de partida.

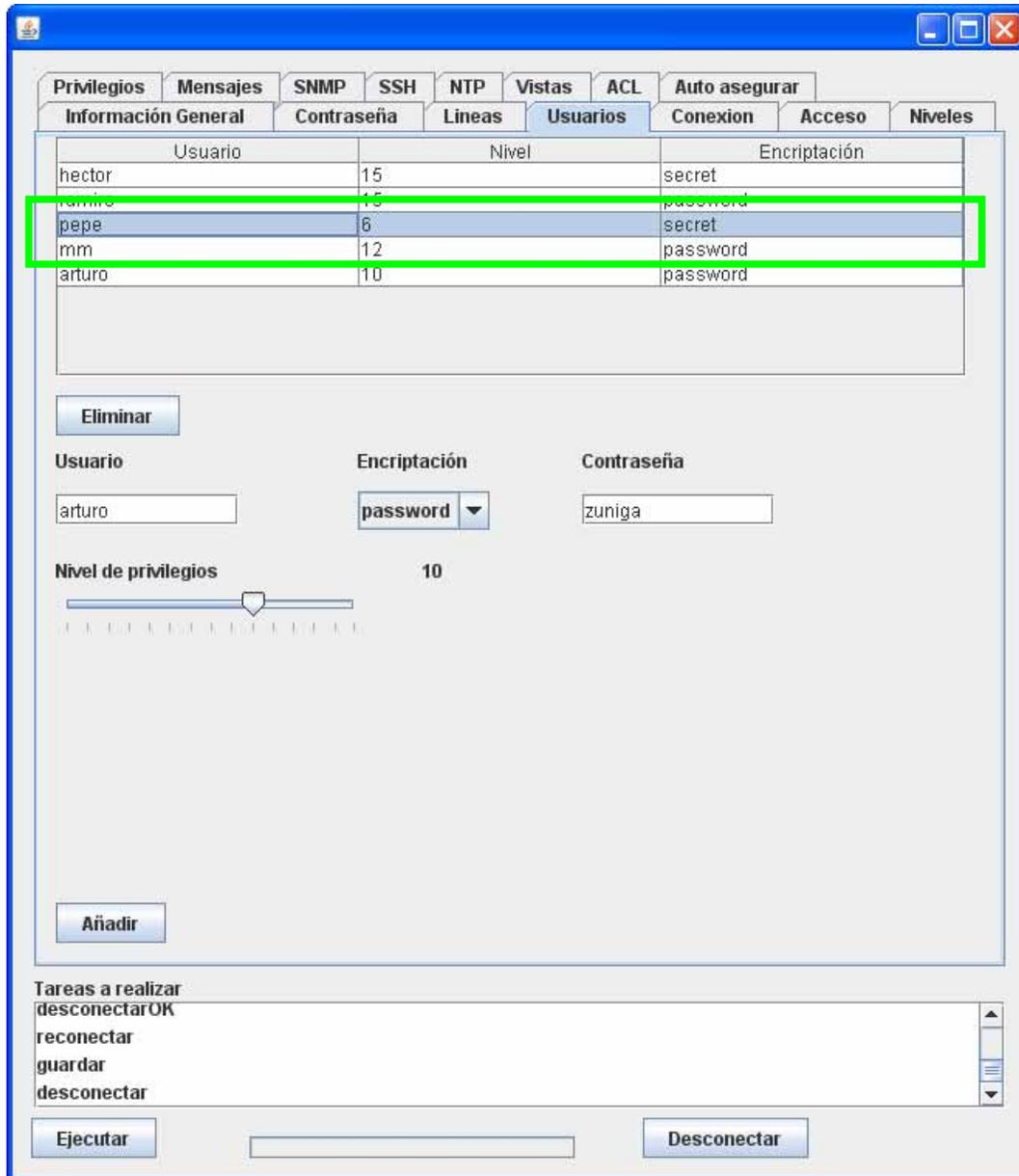
The screenshot shows a network configuration window with a blue title bar and standard window controls. The interface is divided into several sections:

- Navigation Tabs:** Privilegios, Mensajes, SNMP, SSH, NTP, Vistas, ACL, Auto asegurar, Información General, Contraseña, Líneas, Usuarios (selected), Conexión, Acceso, Niveles.
- User List Table:** A table with three columns: Usuario, Nivel, and Encriptación. The row for 'arturo' is highlighted with a green border.
- Form Fields:** Below the table, there are input fields for 'Usuario' (containing 'arturo'), 'Encriptación' (a dropdown menu set to 'password'), and 'Contraseña' (containing 'zuniga').
- Privilege Slider:** A slider labeled 'Nivel de privilegios' is set to '10'.
- Buttons:** 'Eliminar' (highlighted), 'Añadir', 'Ejecutar', and 'Desconectar' are visible.
- Task List:** A list of tasks to perform: desconectarOK, reconectar, guardar, desconectar.

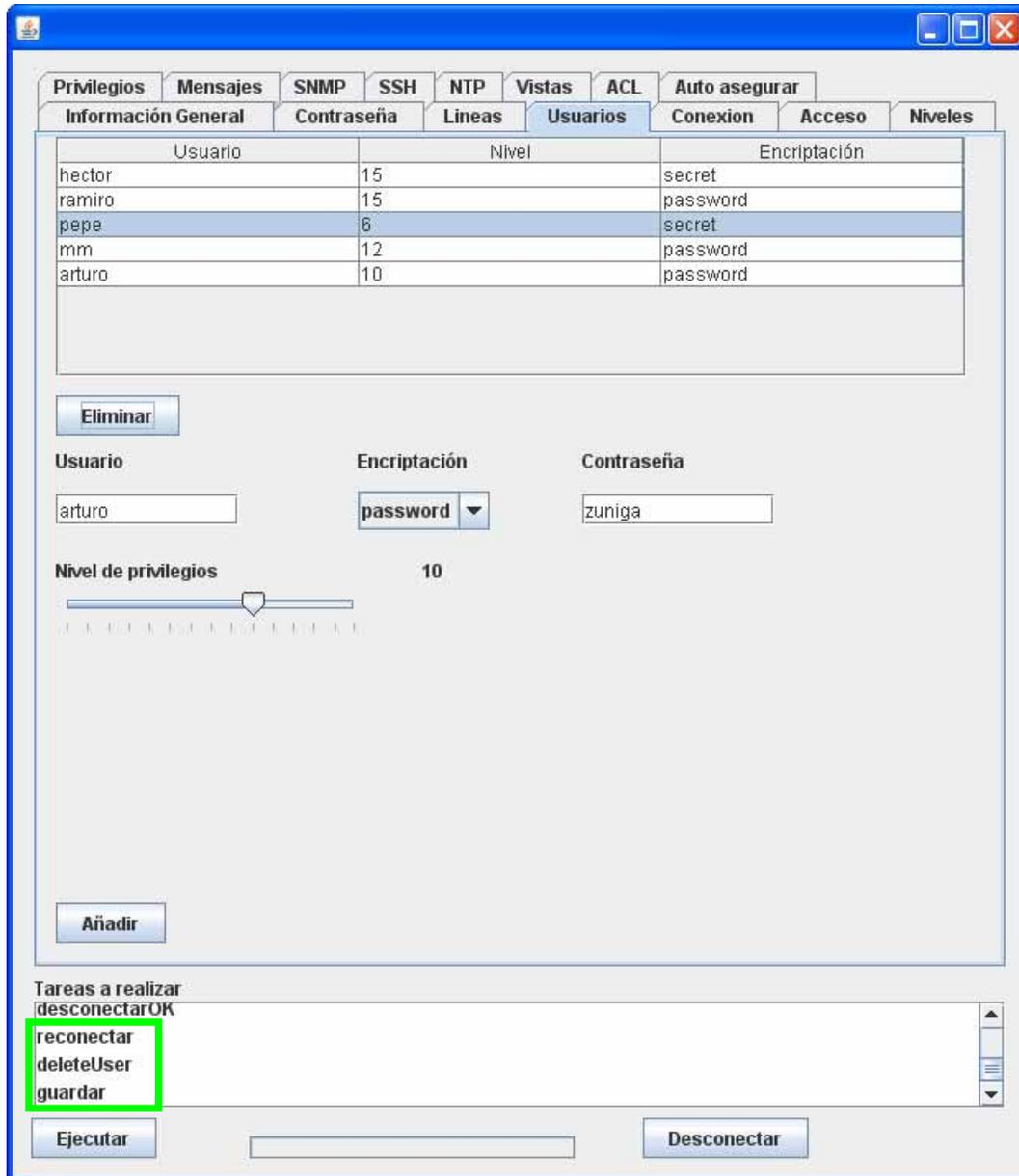
| Usuario | Nivel | Encriptación |
|---------|-------|--------------|
| hector  | 15    | secret       |
| ramiro  | 15    | password     |
| pepe    | 6     | secret       |
| mm      | 12    | password     |
| arturo  | 10    | password     |

Se confirman los mismos datos dentro del Router (ruteador).

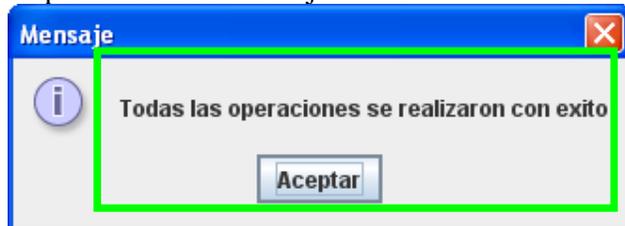




Se presiona el botón de eliminar y automáticamente se añade la tarea a la lista de tareas ejecutarse.



Se presiona el botón de ejecutar.



Se aprecian los cambios en la interfaz.

The screenshot displays a network configuration window with several tabs: Privilegios, Mensajes, SNMP, SSH, NTP, Vistas, ACL, and Auto asegurar. The 'Usuarios' tab is active, showing a table of users and their configurations. Below the table are fields for editing a user's details, including a name field (containing 'arturo'), an encryption dropdown (set to 'password'), a password field (containing 'zuniga'), and a privilege level slider (set to 10). At the bottom, there are buttons for 'Ejecutar' and 'Desconectar', along with a list of tasks to perform: desconectarOK, reconectar, guardar, and desconectar.

| Usuario | Nivel | Encriptación |
|---------|-------|--------------|
| hector  | 15    | secret       |
| ramiro  | 15    | password     |
| mm      | 12    | password     |
| arturo  | 10    | password     |

**Eliminar**

**Usuario**  **Encriptación**  **Contraseña**

**Nivel de privilegios** 10

**Añadir**

**Tareas a realizar**  
desconectarOK  
reconectar  
guardar  
desconectar

**Ejecutar**  **Desconectar**

Se comprueban los cambios en la configuración del Router (ruteador).



Prueba 008: cambiar tiempo válido de conexión sin actividad en una línea.  
Punto de partida.

The screenshot shows a configuration window for a network device. At the top, there are tabs for various services: Privilegios, Mensajes, SNMP, SSH, NTP, Vistas, ACL, and Auto asegurar. Below these are sub-tabs: Información General, Contraseña, Líneas, Usuarios, Conexión, Acceso, and Niveles. The 'Línea' section is active, showing a dropdown menu set to 'vtty 0'. Below this, there are input fields for 'Minutos' (set to 0) and 'Segundos' (set to 0), and an 'Añadir' button. A table lists the configuration for several lines:

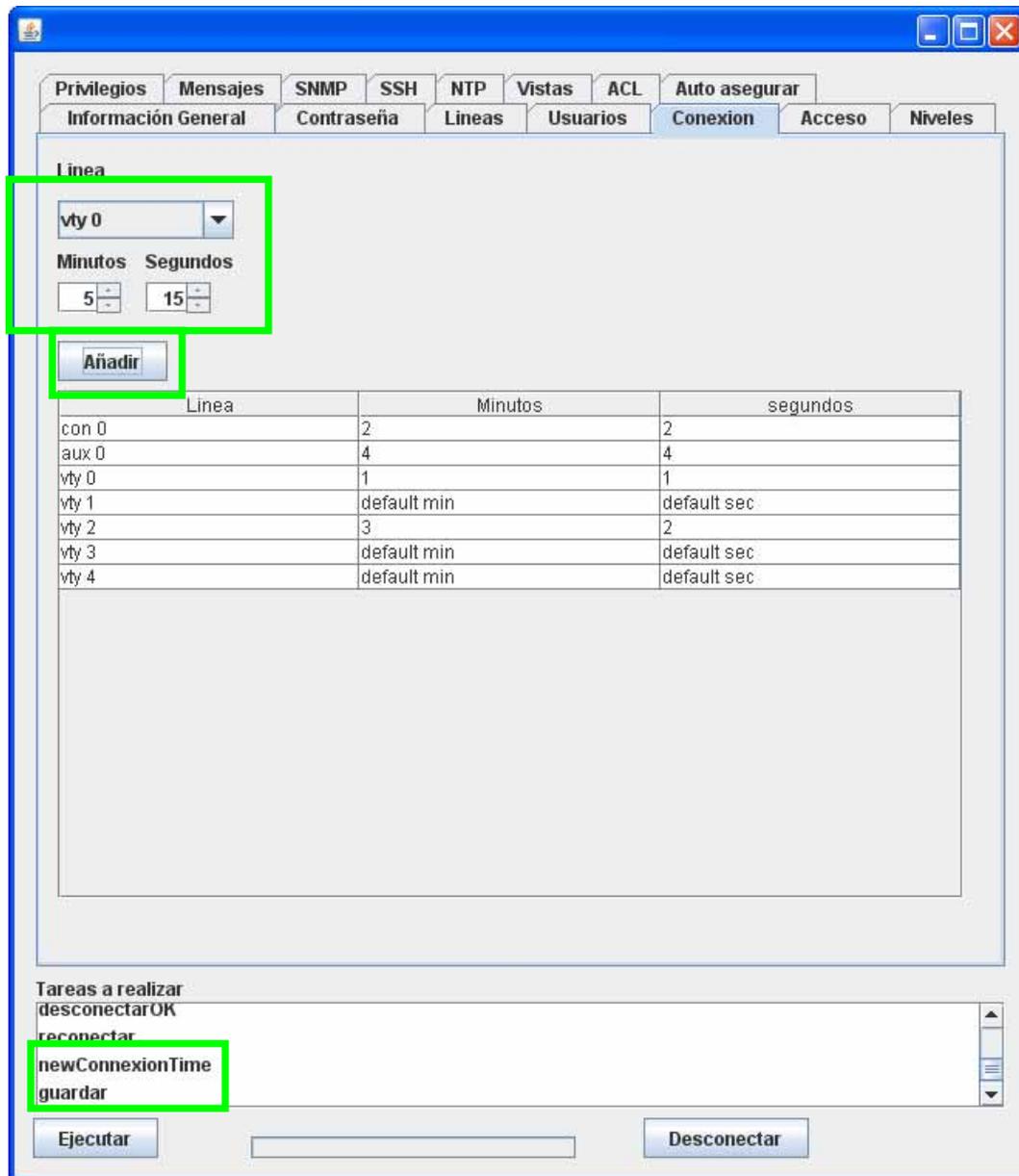
| Línea  | Minutos     | segundos    |
|--------|-------------|-------------|
| con 0  | 2           | 2           |
| aux 0  | 4           | 4           |
| vtty 0 | 1           | 1           |
| vtty 1 | default min | default sec |
| vtty 2 | 3           | 2           |
| vtty 3 | default min | default sec |
| vtty 4 | default min | default sec |

At the bottom, there is a 'Tareas a realizar' section with a list of actions: desconectarOK, reconectar, guardar, and desconectar. Below this list are 'Ejecutar' and 'Desconectar' buttons.

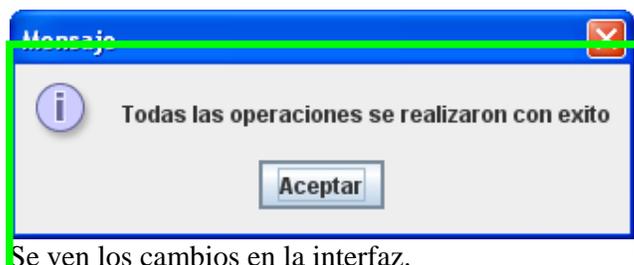
Se comprueban los mismos datos dentro de la configuración del Router (ruteador).

```
Dynamips(0): R1, Console port
snmp-server view vista cisco included
snmp-server community lectura view vista RO
snmp-server community escritura view vista RW
snmp-server enable traps tty
!
control-plane
!
privilege exec all level 5 ping
privilege exec level 2 auto
privilege exec level 4 debug
!
line con 0
exec-timeout 2 2
line aux 0
exec-timeout 4 4
line vty 0
exec-timeout 1 1
password linea
line vty 1
password linea
line vty 2
exec-timeout 3 2
password jojojo
line vty 3
password hola
line vty 4
password adios
parser view vista1
secret 5 $1$TVOB$E5HmQ3ZnI4zGeF1S2ch2C.
commands exec include all ping
commands exec include show running-config
commands exec include show
commands exec exclude debug
!
!
ntp server 10.1.1.3 prefer
ntp server 10.1.1.2 prefer
end
Router#
```

Se escoge una línea. También se seleccionan los minutos y segundos. Se presiona el botón de añadir lo cual agrega automáticamente la tarea a la lista de tareas ejecutarse.



Se presiona el botón de ejecutar.



The screenshot shows a configuration window for a network device. At the top, there are several tabs: Privilegios, Mensajes, SNMP, SSH, NTP, Vistas, ACL, Auto asegurar, Información General, Contraseña, Líneas, Usuarios, Conexion, Acceso, and Niveles. The 'Conexion' tab is selected. Below the tabs, the 'Línea' section is active, showing a dropdown menu with 'con 0' selected. Underneath, there are input fields for 'Minutos' (set to 5) and 'Segundos' (set to 15). An 'Añadir' button is located below these fields. A table displays the configuration for various lines:

| Línea | Minutos     | segundos    |
|-------|-------------|-------------|
| con 0 | 2           | 2           |
| aux 0 | 4           | 4           |
| vty 0 | 5           | 15          |
| vty 1 | default min | default sec |
| vty 2 | 3           | 2           |
| vty 3 | default min | default sec |
| vty 4 | default min | default sec |

At the bottom of the window, there is a section titled 'Tareas a realizar' with a list of actions: desconectarOK, reconectar, guardar, and desconectar. Below this list are two buttons: 'Ejecutar' and 'Desconectar', with an empty text input field between them.

Se comprueban los datos dentro de la configuración del Router (ruteador).

```
Dynamips(0): R1, Console port
snmp-server view vista cisco included
snmp-server community lectura view vista RO
snmp-server community escritura view vista RW
snmp-server enable traps tty
!
control-plane
!
privilege exec all level 5 ping
privilege exec level 2 auto
privilege exec level 4 debug
!
line con 0
exec-timeout 2 2
line aux 0
exec-timeout 4 4
line vty 0
exec-timeout 5 15
password linea
line vty 1
password linea
line vty 2
exec-timeout 3 2
password jojojo
line vty 3
password hola
line vty 4
password adios
parser view vista1
secret 5 $1$TVOB$E5HmQ3ZnI4zGeF1S2ch2C.
commands exec include all ping
commands exec include show running-config
commands exec include show
commands exec exclude debug
!
!
ntp server 10.1.1.3 prefer
ntp server 10.1.1.2 prefer
end
Router#
```

Prueba 009: cambio de número máximo de intentos de acceso  
Punto de partida.

The screenshot shows a web-based configuration interface for a network device. At the top, there are several tabs: Privilegios, Mensajes, SNMP, SSH, NTP, Vistas, ACL, Auto asegurar, Información General, Contraseña, Líneas, Usuarios, Conexion, Acceso, and Niveles. The 'Acceso' tab is currently selected. Below the tabs, there is a label 'Número de intentos establecidos:' followed by a spin box containing the number '2'. Below the spin box is an 'Añadir' button. At the bottom of the main configuration area, there is a section titled 'Tareas a realizar' with a list of tasks: desconectarOK, reconectar, guardar, and desconectar. Below this list are two buttons: 'Ejecutar' and 'Desconectar', with an empty text input field between them.

Se confirman los datos dentro del Router (ruteador).

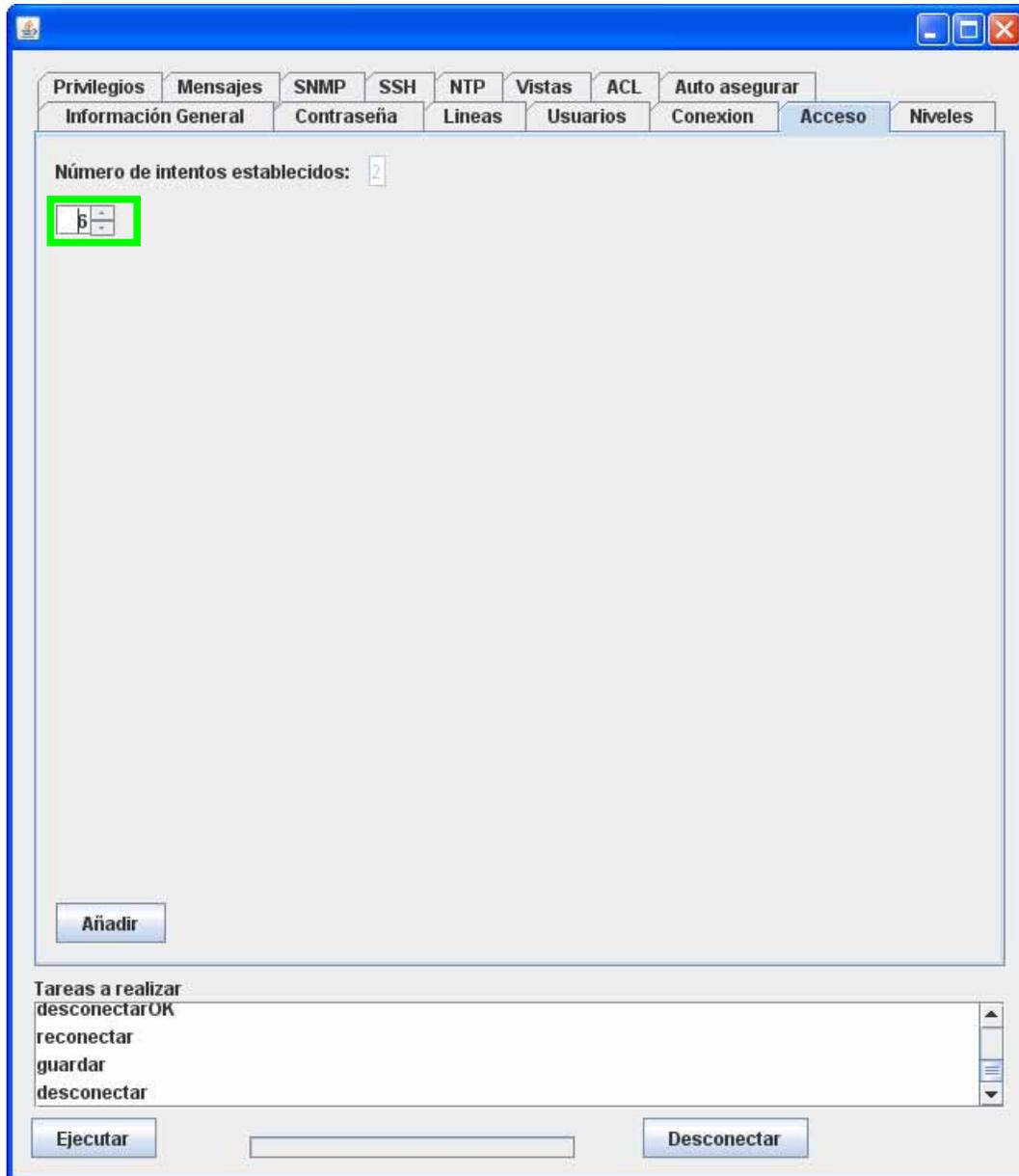
```
Dynamips(0): R1, Console port

Router>enable
Password:
Router#show run
^
% Invalid input detected at '^' marker.

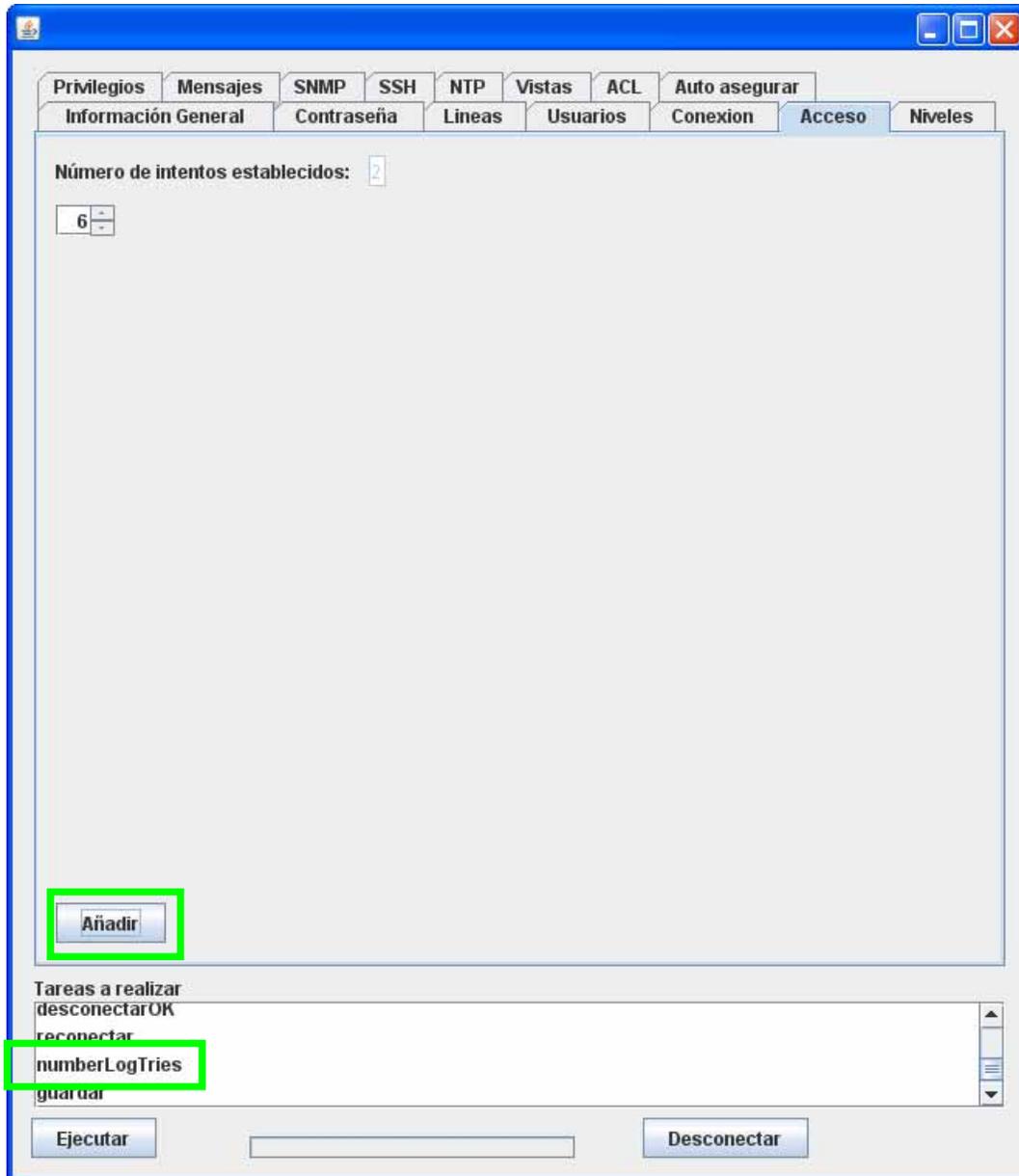
Router#show run
Building configuration...

Current configuration : 2109 bytes
!
version 12.4
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname Router
!
boot-start-marker
boot-end-marker
!
security authentication failure rate 2 log
enable secret level 5 5 $1$3P34$W3W,33$RP7pj/edU0
enable secret level 7 5 $1$d3Sf$7gOgzcp3/m1veVMct4K01
enable secret level 9 5 $1$jaK/$0nB1CJ9mFPu4zs6KbNzo91
enable secret level 12 5 $1$HVe9$Mcv9aWUUruxtFkw9UJq4e.
enable password cisco
!
aaa new-model
!
!
!
aaa session-id common
memory-size iomem 15
ip cef
!
!
ip auth-proxy max-nodata-conns 3
```

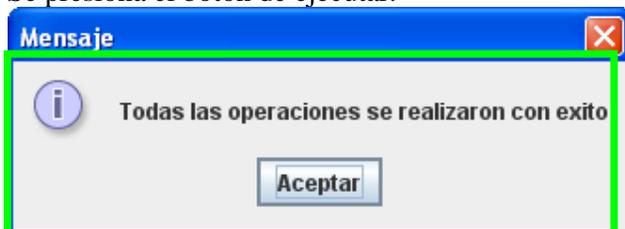
Se establece un nuevo número de intentos.



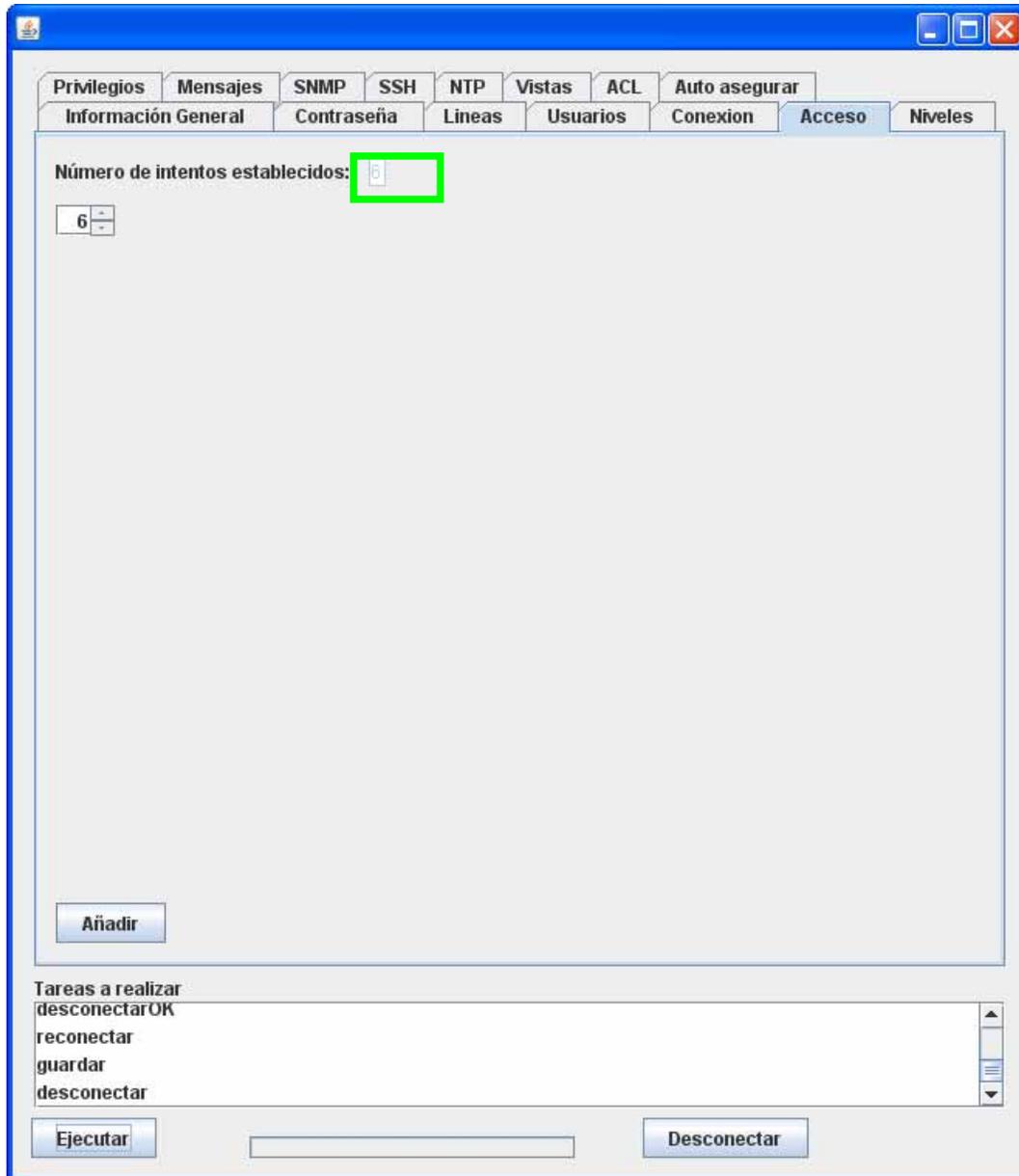
Se presiona el botón de añadir lo cual agrega la tarea a la lista de tareas ejecutarse.



Se presiona el botón de ejecutar.



Se aprecian los cambios en la interfaz.



Se comprueban los cambios dentro de la configuración del Router (ruteador).

```
Dynamips(0): R1, Console port
parser view vista1
secret 5 $1$TVOB$E5HmQ3ZnI4zGeF1S2ch2C.
commands exec include all ping
commands exec include show running-config
commands exec include show
commands exec exclude debug
!
!
ntp server 10.1.1.3 prefer
ntp server 10.1.1.2 prefer
end

Router#
*Mar 1 01:06:29.991: %SYS-5-CONFIG_I: Configured from console by hector on vty0
(10.1.1.2)show run
Building configuration...

Current configuration : 2109 bytes
!
version 12.4
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname Router
!
boot-start-marker
boot-end-marker
!
security authentication failure rate 6 log
enable secret level 5 5 $1$cGRI$wbtOz1Ww37ubRP5pj/edU0
enable secret level 7 5 $1$d38f$7gOgzcp3/m1veVMct4K01
enable secret level 9 5 $1$jaK/$OnB1CJ9mFPu4zs6KbNzo91
enable secret level 12 5 $1$HVe9$Mcv9aWUUruxtFkw9UJq4e.
enable password cisco
!
aaa new-model
!
!
--More--
```

Prueba 010: habilitar contraseña para diferentes niveles de acceso.  
Punto de partida.

The screenshot shows a configuration window with several tabs: Privilegios, Mensajes, SNMP, SSH, NTP, Vistas, ACL, Auto asegurar, Información General, Contraseña, Lineas, Usuarios, Conexion, Acceso, and Niveles. The 'Niveles' tab is active, displaying a table with two columns: 'Niveles' and 'Secreto'. The table lists levels 1 through 15 and their password status. A green border highlights the table area.

| Niveles  | Secreto       |
|----------|---------------|
| level 1  | habilitado    |
| level 2  | deshabilitado |
| level 3  | deshabilitado |
| level 4  | deshabilitado |
| level 5  | habilitado    |
| level 6  | deshabilitado |
| level 7  | habilitado    |
| level 8  | deshabilitado |
| level 9  | habilitado    |
| level 10 | deshabilitado |
| level 11 | deshabilitado |
| level 12 | habilitado    |
| level 13 | deshabilitado |
| level 14 | deshabilitado |
| level 15 | deshabilitado |

Below the table is an 'Añadir' button. At the bottom of the window, there is a 'Tareas a realizar' section with a list of tasks: desconectarOK, reconectar, guardar, and desconectar. There are also 'Ejecutar' and 'Desconectar' buttons at the bottom.

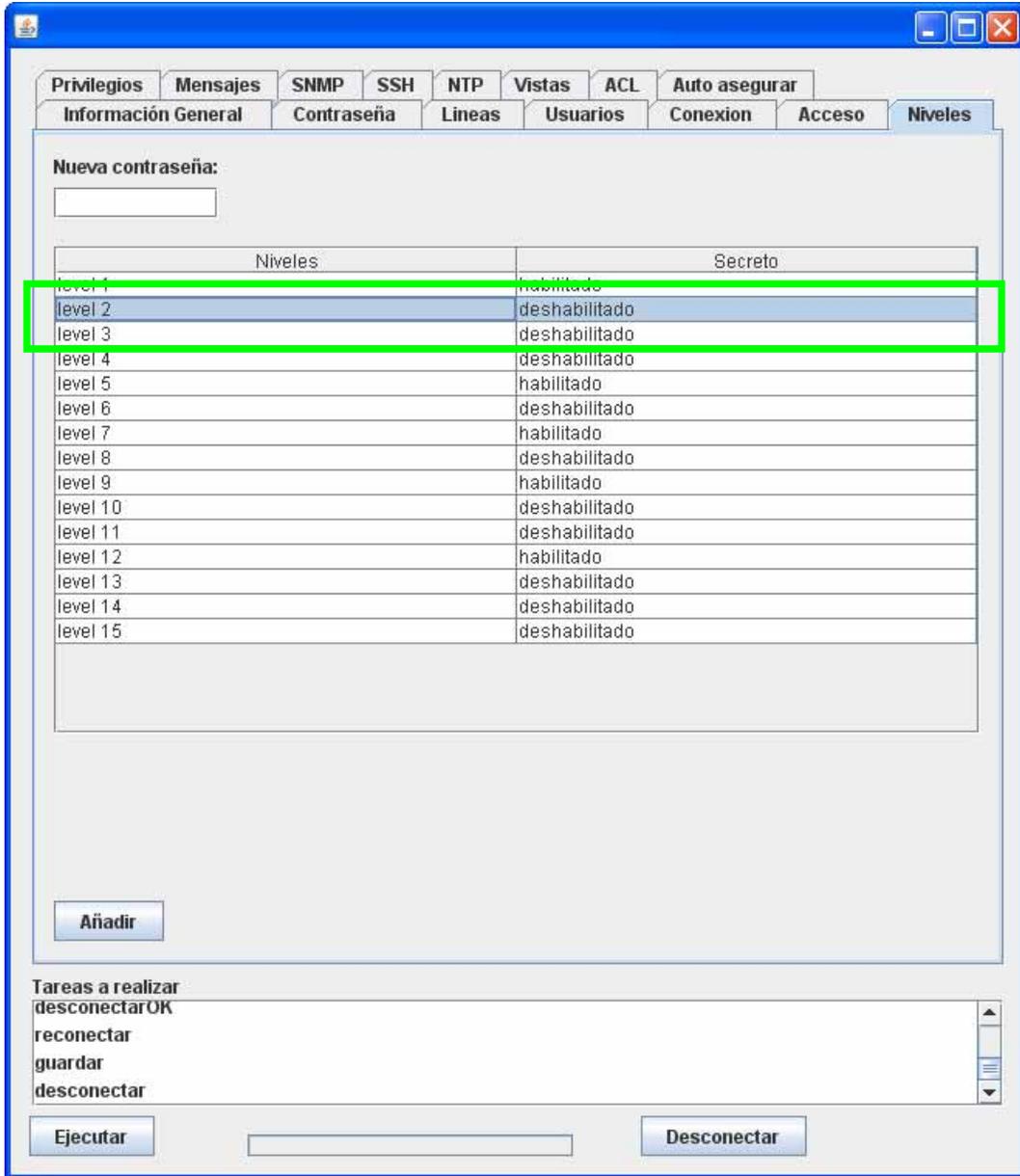
Se confirma la misma configuración dentro del Router (ruteador).

```
Dynamips(0): R1, Console port
parser view vista1
secret 5 $1$TVOB$E5HmQ3ZnI4zGeF1S2ch2C.
commands exec include all ping
commands exec include show running-config
commands exec include show
commands exec exclude debug
!
!
ntp server 10.1.1.3 prefer
ntp server 10.1.1.2 prefer
end

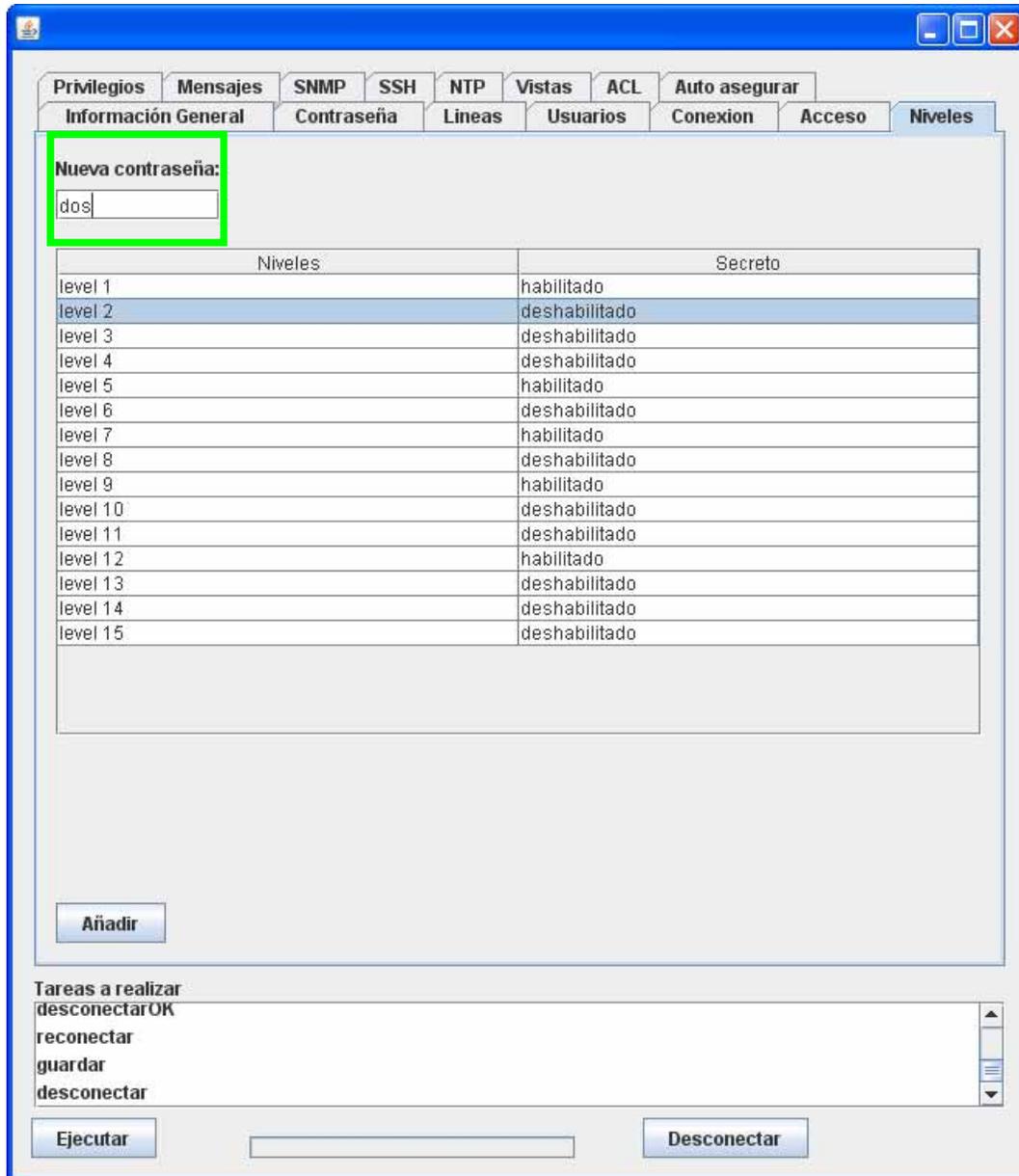
Router#
*Mar 1 01:06:29.991: %SYS-5-CONFIG_I: Configured from console by hector on vty0
(10.1.1.2)show run
Building configuration...

Current configuration : 2109 bytes
!
version 12.4
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname Router
!
boot-start-marker
boot-end-marker
!
security authentication failure rate 6 log
enable secret level 5 5 $1$sGRI$ybtQz1Vy3ZubRP7pj/edU0
enable secret level 7 5 $1$d3Sf$7gOgzcp3/m1veVMct4K01
enable secret level 9 5 $1$jaK/$OnB1CJ9mFPu4zs6KbNzo91
enable secret level 12 5 $1$HVe9$Mcv9aWUruytFkw9UJq4e.
enable password cisco
!
aaa new-model
!
!
--More--
```

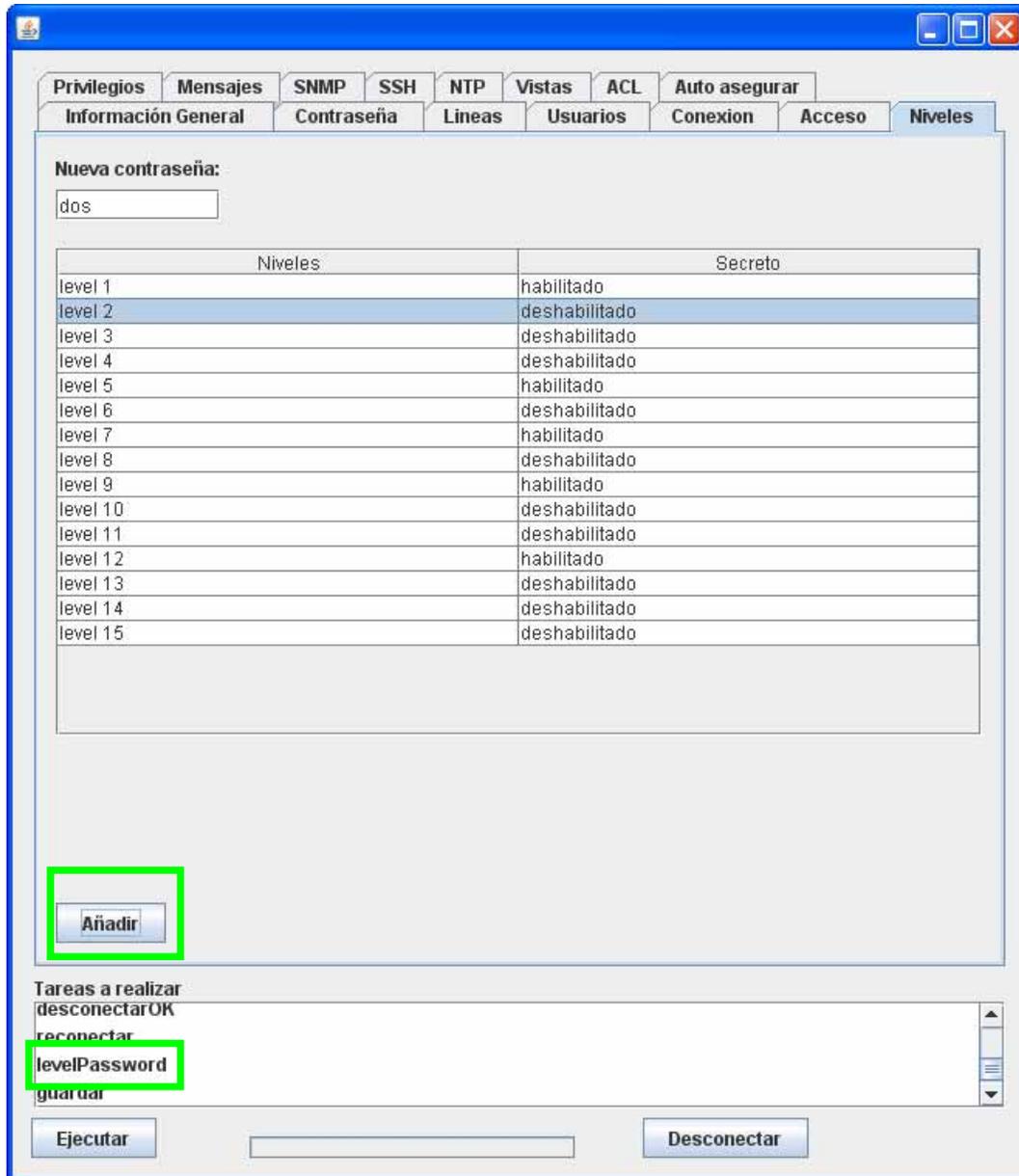
Se selecciona un nivel.



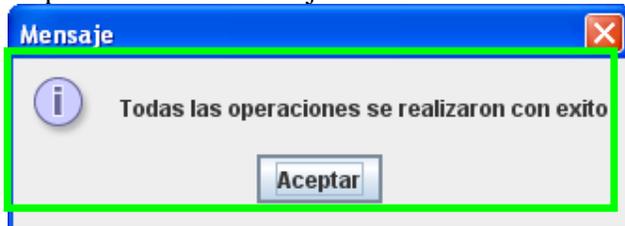
Se le establece una nueva contraseña.



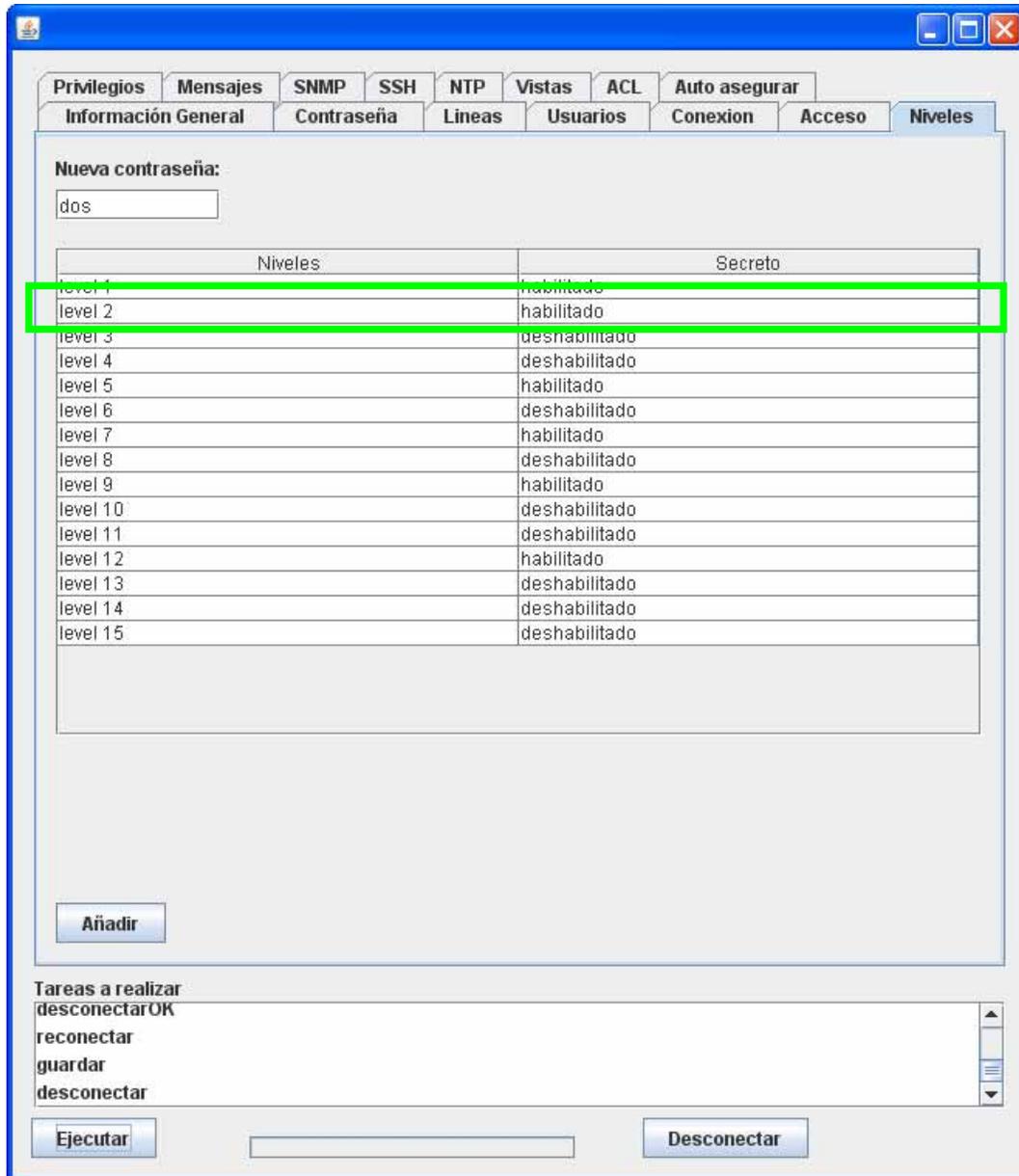
Se presiona el botón de añadir lo cual agrega la tarea a la lista de tareas ejecutarse.



Se presiona el botón de ejecutar.



Se ven los cambios en la interfaz.



Se comprueban los cambios dentro de la configuración del Router (ruteador).

```
Dynamips(0): R1, Console port
!
security authentication failure rate 6 log
enable secret level 5 5 $1$sGRI$ybtQz1Vy3ZubRP7pj/edU0
enable secret level 7 5 $1$d3Sf$7gOgzcp3/m1veVMct4K01
enable secret level 9 5 $1$jaK/$0nB1CJ9mFPu4zs6KbNzo91
enable secret level 12 5 $1$HVe9$Mcv9aWUruytFkw9UJq4e.
enable password cisco
!
aaa new-model
!
!
--More--
*Mar  1 01:08:16.375: %SYS-5-CONFIG_I: Configured from console by hector on vty0
(
)
Router#show run
Building configuration...

Current configuration : 2164 bytes
!
version 12.4
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname Router
!
boot-start-marker
boot-end-marker
!
security authentication failure rate 6 log
enable secret level 2 5 $1$LrC8$Mu9iuHq1JFGGB07R7DmTW.
enable secret level 5 5 $1$sGRI$ybtQz1Vy3ZubRP7pj/edU0
enable secret level 7 5 $1$d3Sf$7gOgzcp3/m1veVMct4K01
enable secret level 9 5 $1$jaK/$0nB1CJ9mFPu4zs6KbNzo91
enable secret level 12 5 $1$HVe9$Mcv9aWUruytFkw9UJq4e.
enable password cisco
!
aaa new-model
!
--More--
```

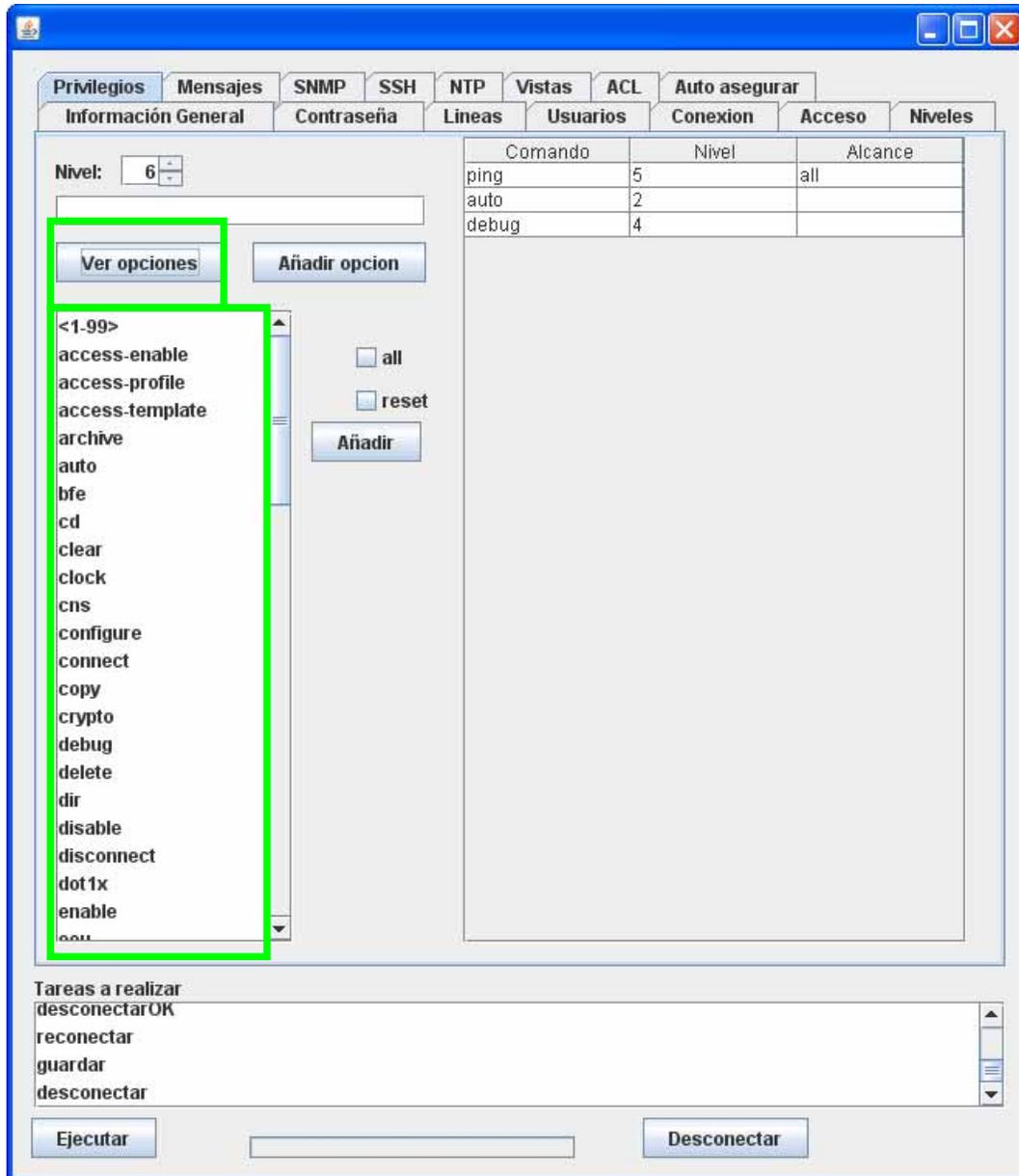
Prueba 011: establecer nivel de privilegio para un comando.  
Se selecciona nivel al cual se desea establecer el comando.

The screenshot shows a web-based configuration interface for a network device. The 'Niveles' (Privileges) tab is active. The 'Nivel' field is highlighted with a green box and contains the value '6'. Below it are buttons for 'Ver opciones' and 'Añadir opcion'. To the right, a table lists commands and their levels:

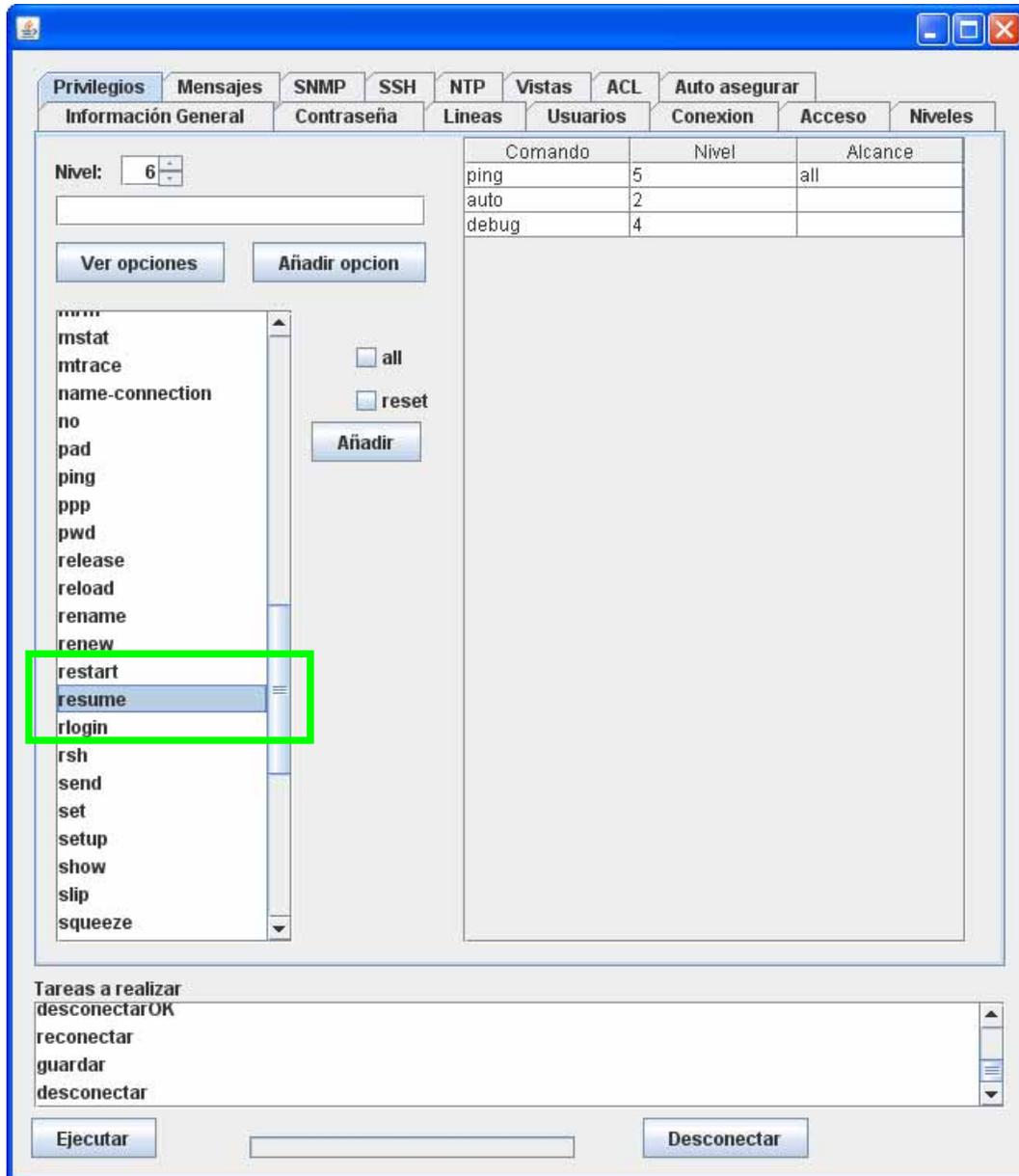
| Comando | Nivel | Alcance |
|---------|-------|---------|
| ping    | 5     | all     |
| auto    | 2     |         |
| debug   | 4     |         |

Below the table are checkboxes for 'all' and 'reset', and an 'Añadir' button. At the bottom, there is a 'Tareas a realizar' section with a list of tasks: desconectarOK, reconectar, guardar, and desconectar. There are also 'Ejecutar' and 'Desconectar' buttons.

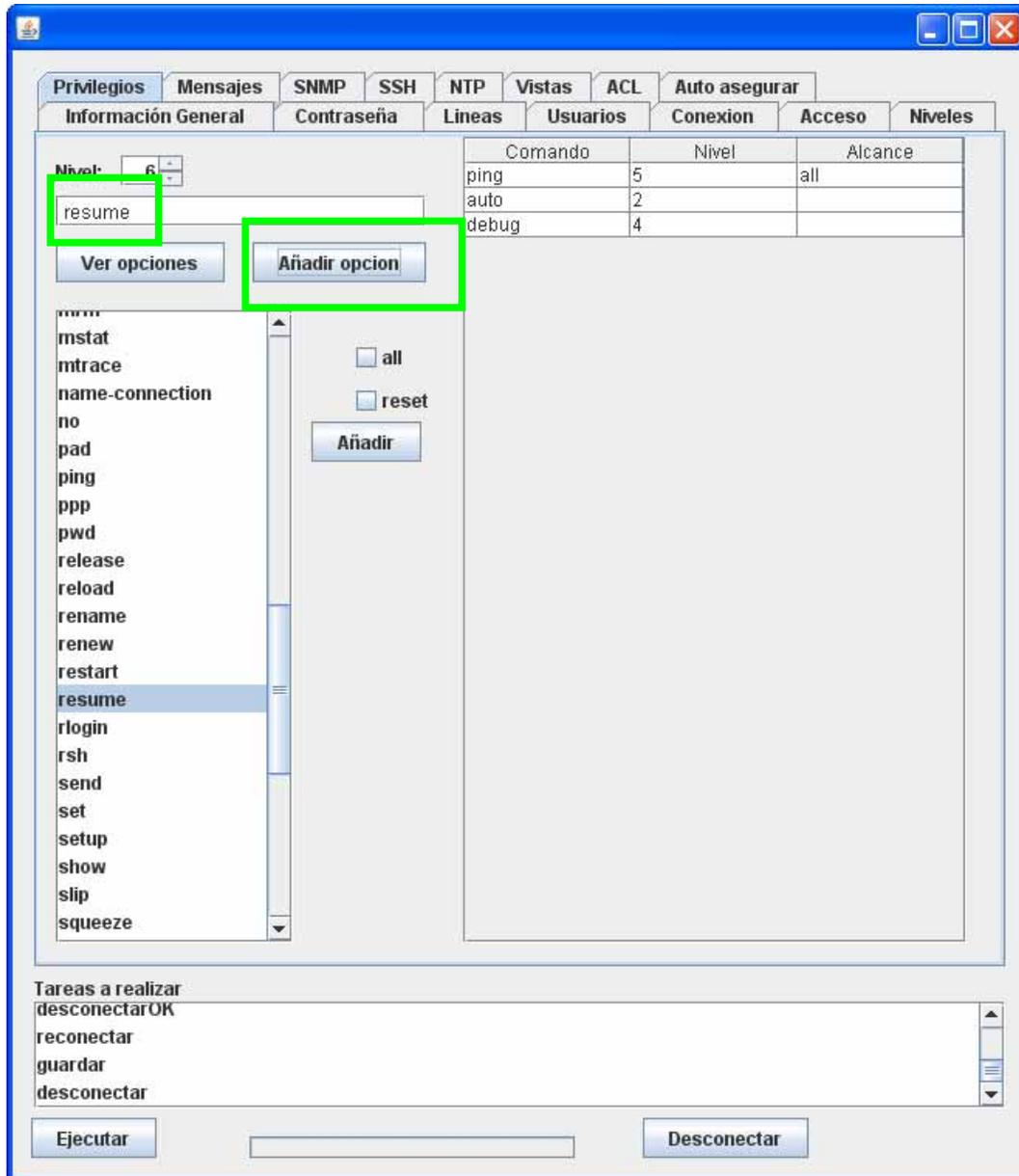
Se presiona el botón de ver opciones.



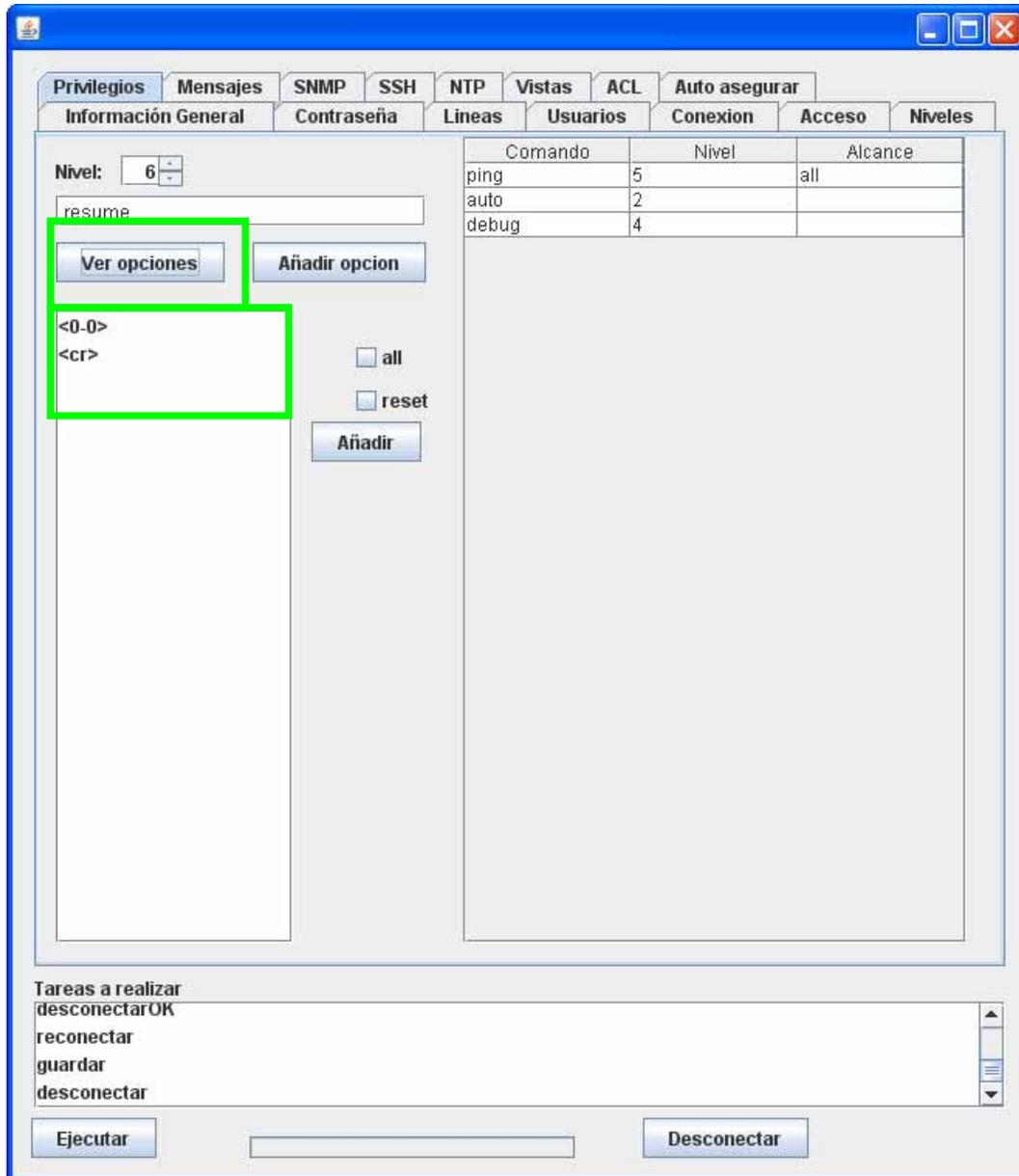
Se selecciona una opción.



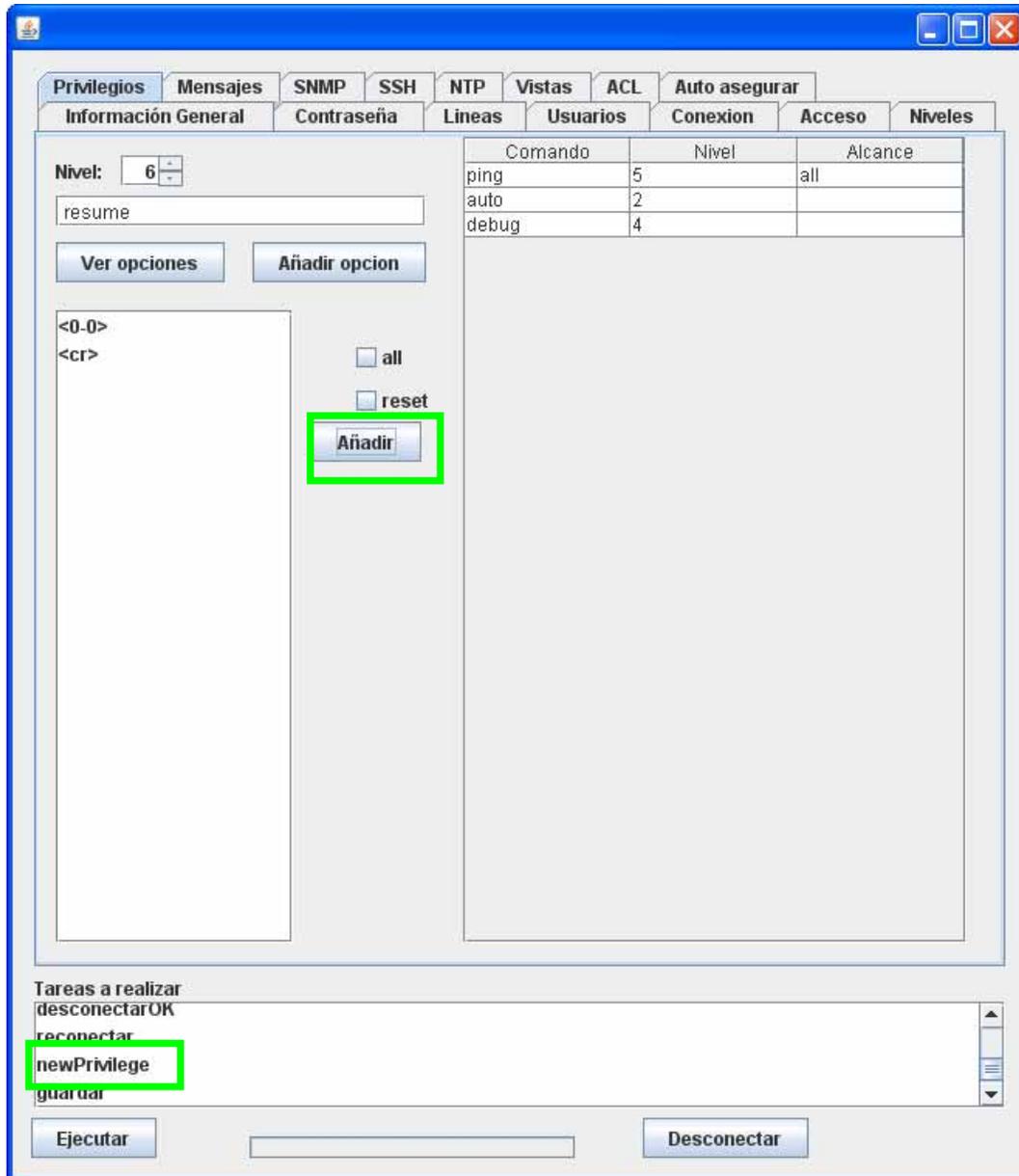
Se presiona el botón de añadir opción.



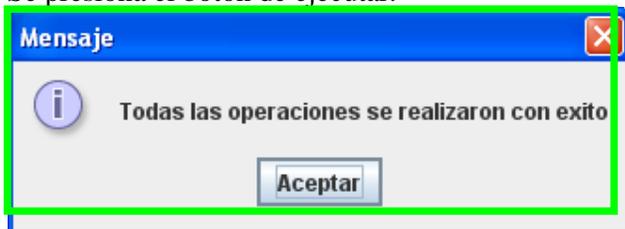
Se presiona de nuevo el botón de ver opciones.



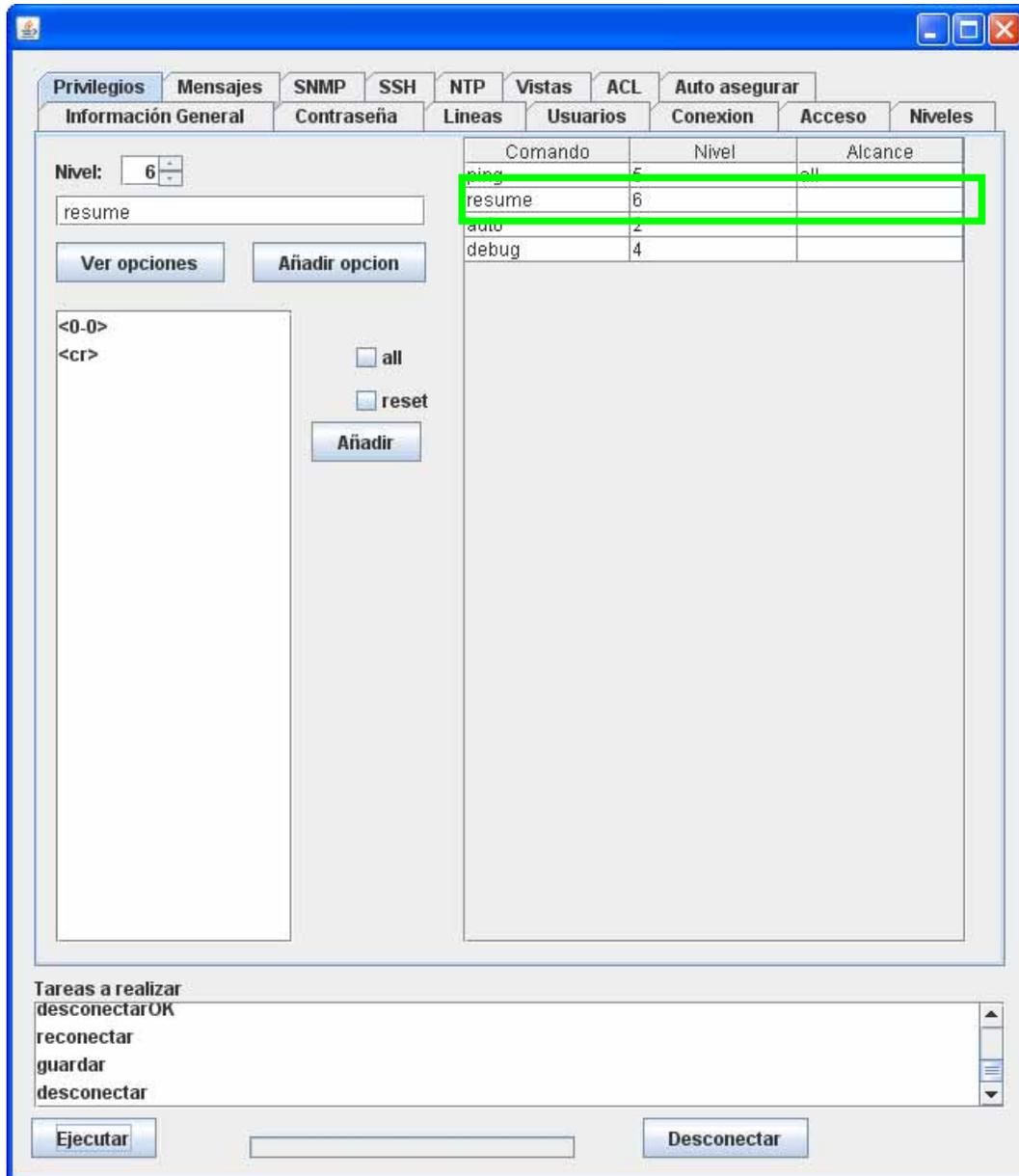
Se presiona el botón de añadir para agregar la tarea a la lista de tareas a ejecutarse.



Se presiona el botón de ejecutar.



Se aprecian los cambios en la interfaz.



Se comprueban los cambios dentro de la configuración del Router (ruteador).

```
Dynamips(0): R1, Console port
interface FastEthernet0
 ip address 10.1.1.1 255.255.255.0
 speed auto
!
interface Serial0
 no ip address
 shutdown
!
interface Serial1
 no ip address
 shutdown
!
ip forward-protocol nd
!
no ip http server
no ip http secure-server
!
access-list 10 permit 10.1.1.2
access-list 10 permit 10.1.1.3
access-list 10 permit 10.1.1.4
access-list 10 deny 10.1.1.5
access-list 10 deny any log
snmp-server view vista iso.* included
snmp-server view vista system included
snmp-server view vista cisco included
snmp-server community lectura view vista RO
snmp-server community escritura view vista RW
snmp-server enable traps tty
!
control-plane
!
privilege exec level 5 ping
privilege exec level 6 resume
privilege exec level 2 auto
privilege exec level 4 debug
!
line con 0
 exec-timeout 2 2
line aux 0
--More--
```

Prueba 012: restablecer el privilegio de un comando a su nivel original  
Punto de partida

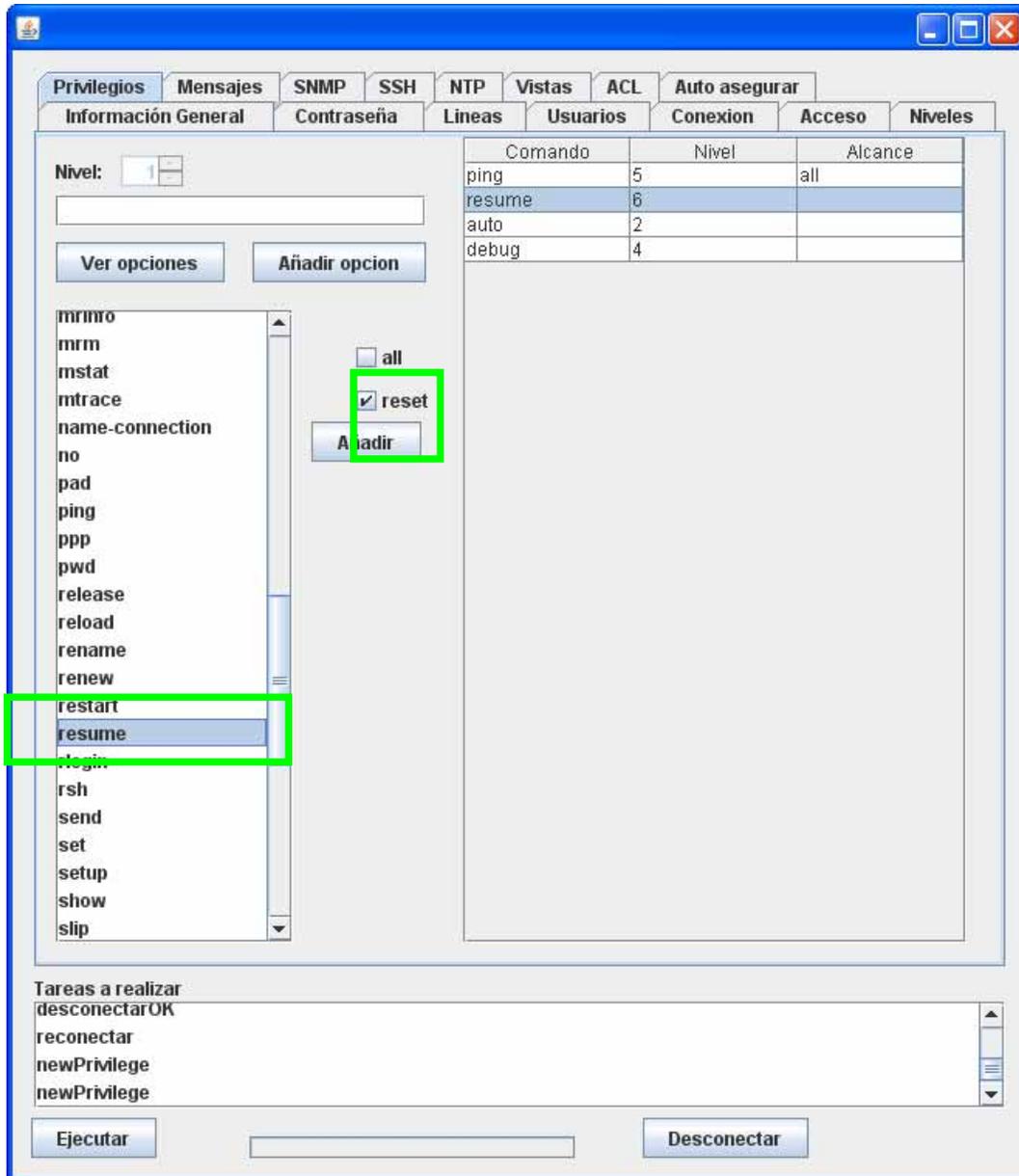
The screenshot shows a network configuration interface with a blue border. At the top, there are several tabs: Privilegios, Mensajes, SNMP, SSH, NTP, Vistas, ACL, and Auto asegurar. Below these, there are sub-tabs: Información General, Contraseña, Líneas, Usuarios, Conexión, Acceso, and Niveles. The 'Niveles' sub-tab is active. On the left, there is a 'Nivel:' field with a dropdown menu showing '1'. Below it are buttons for 'Ver opciones' and 'Añadir opción'. A list of commands is shown in a scrollable area, with 'all' and 'reset' options and an 'Añadir' button. On the right, a table lists commands and their privilege levels. The table is highlighted with a green border. Below the table, there is a 'Tareas a realizar' section with a list of tasks: desconectarOK, reconectar, guardar, and desconectar. At the bottom, there are buttons for 'Ejecutar' and 'Desconectar'.

| Comando | Nivel | Acceso |
|---------|-------|--------|
| ping    | 5     | all    |
| resume  | 6     |        |
| auto    | 2     |        |
| debug   | 4     |        |

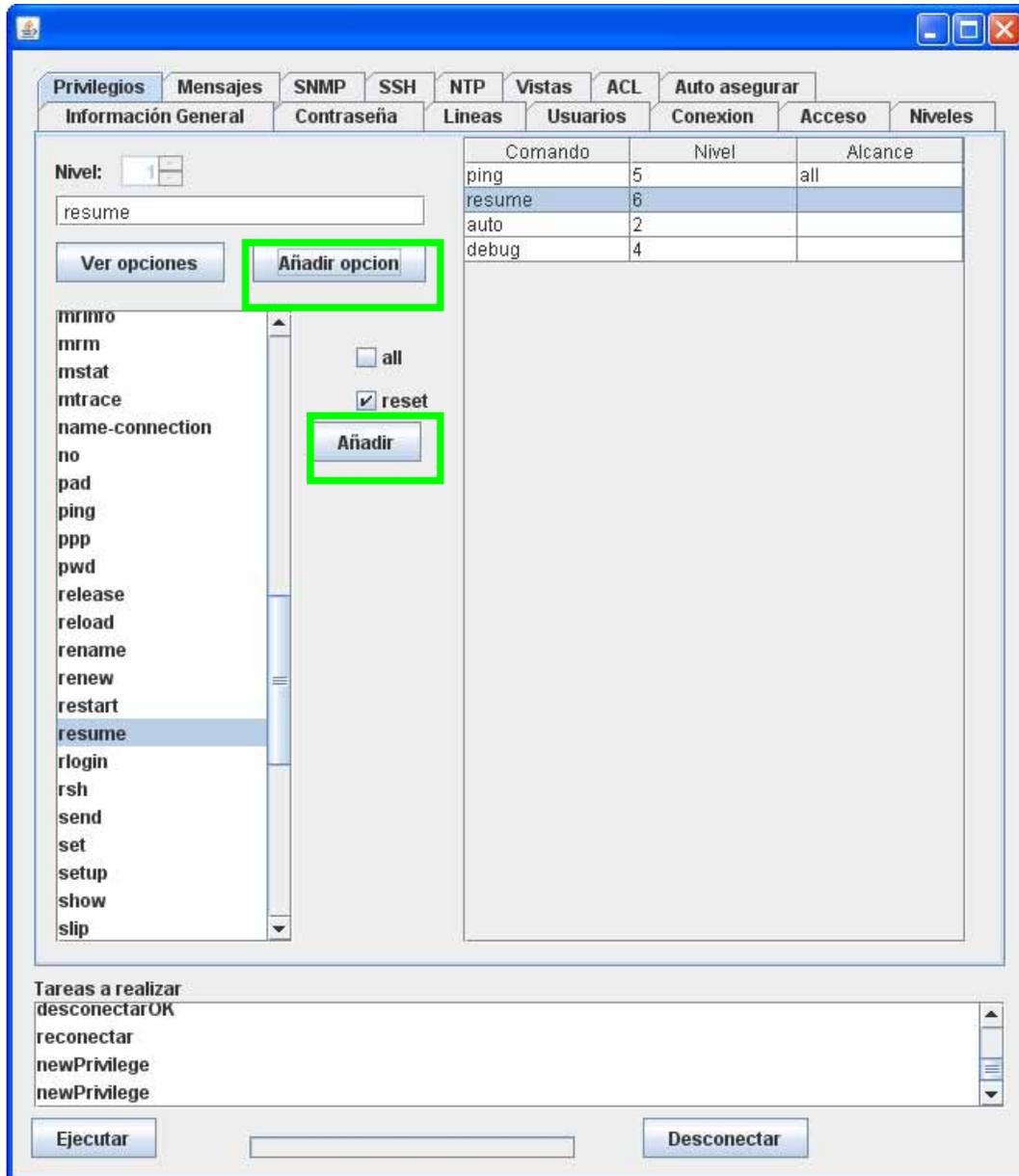
Se confirman los mismos datos dentro de la configuración del Router (ruteador).

```
Dynamips(0): R1, Console port
interface FastEthernet0
 ip address 10.1.1.1 255.255.255.0
 speed auto
!
interface Serial0
 no ip address
 shutdown
!
interface Serial1
 no ip address
 shutdown
!
ip forward-protocol nd
!
no ip http server
no ip http secure-server
!
access-list 10 permit 10.1.1.2
access-list 10 permit 10.1.1.3
access-list 10 permit 10.1.1.4
access-list 10 deny 10.1.1.5
access-list 10 deny any log
snmp-server view vista iso.* included
snmp-server view vista system included
snmp-server view vista cisco included
snmp-server community lectura view vista RO
snmp-server community escritura view vista RW
snmp-server enable traps tty
!
control-plane
!
privilege exec all level 5 ping
privilege exec level 6 resume
privilege exec level 2 auto
privilege exec level 4 debug
!
line con 0
 exec-timeout 2 2
line aux 0
--More--
```

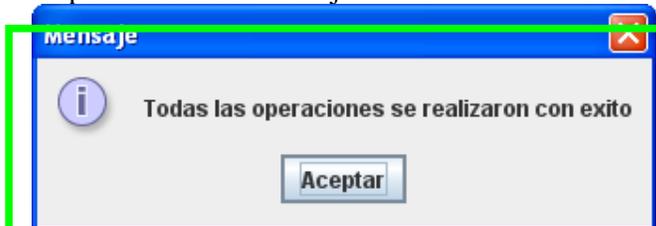
Se selecciona una opción. Y se selecciona el botón de restablecer (reset).



Se aprieta el botón de añadir opción. Y el botón de añadir.



Se presiona el botón de ejecutar.



Se ven los cambios en la interfaz.

The screenshot shows a configuration window for a router with several tabs: Privilegios, Mensajes, SNMP, SSH, NTP, Vistas, ACL, and Auto asegurar. The 'Privilegios' tab is active, showing sub-tabs for Información General, Contraseña, Líneas, Usuarios, Conexión, Acceso, and Niveles. The 'Niveles' sub-tab is selected, displaying a table of command levels and reachability. A green box highlights the table content.

| Comando | Nivel | Alcance |
|---------|-------|---------|
| ping    | 5     | all     |
| auto    | 2     |         |
| debug   | 4     |         |

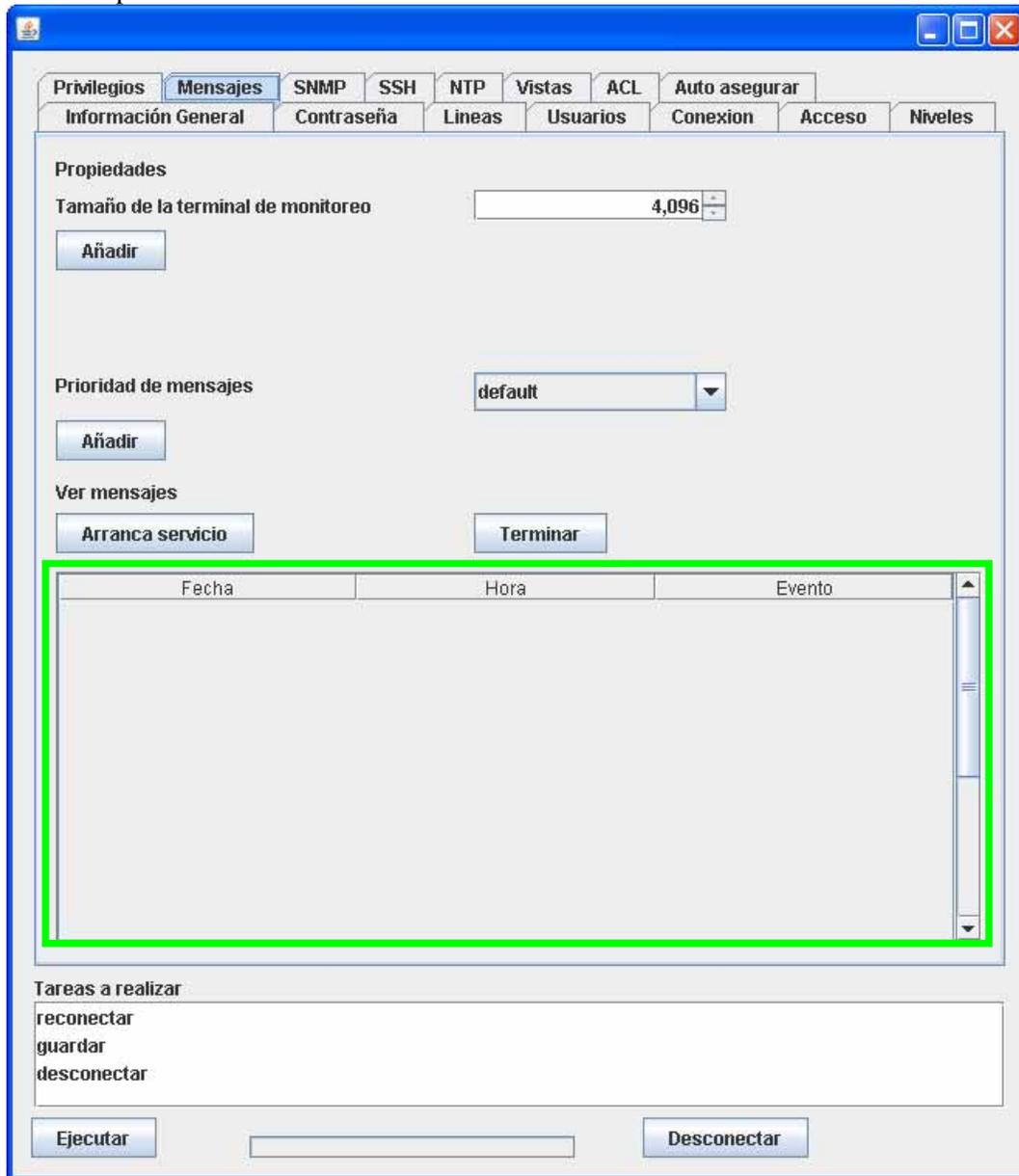
Below the table, there is a list of commands on the left and a list of options on the right. The 'resume' command is selected in the list. The options include 'all' (unchecked) and 'reset' (checked). An 'Añadir' button is present next to the options.

At the bottom, there is a section for 'Tareas a realizar' with a list of tasks: desconectarOK, reconectar, guardar, and desconectar. There are 'Ejecutar' and 'Desconectar' buttons at the very bottom.

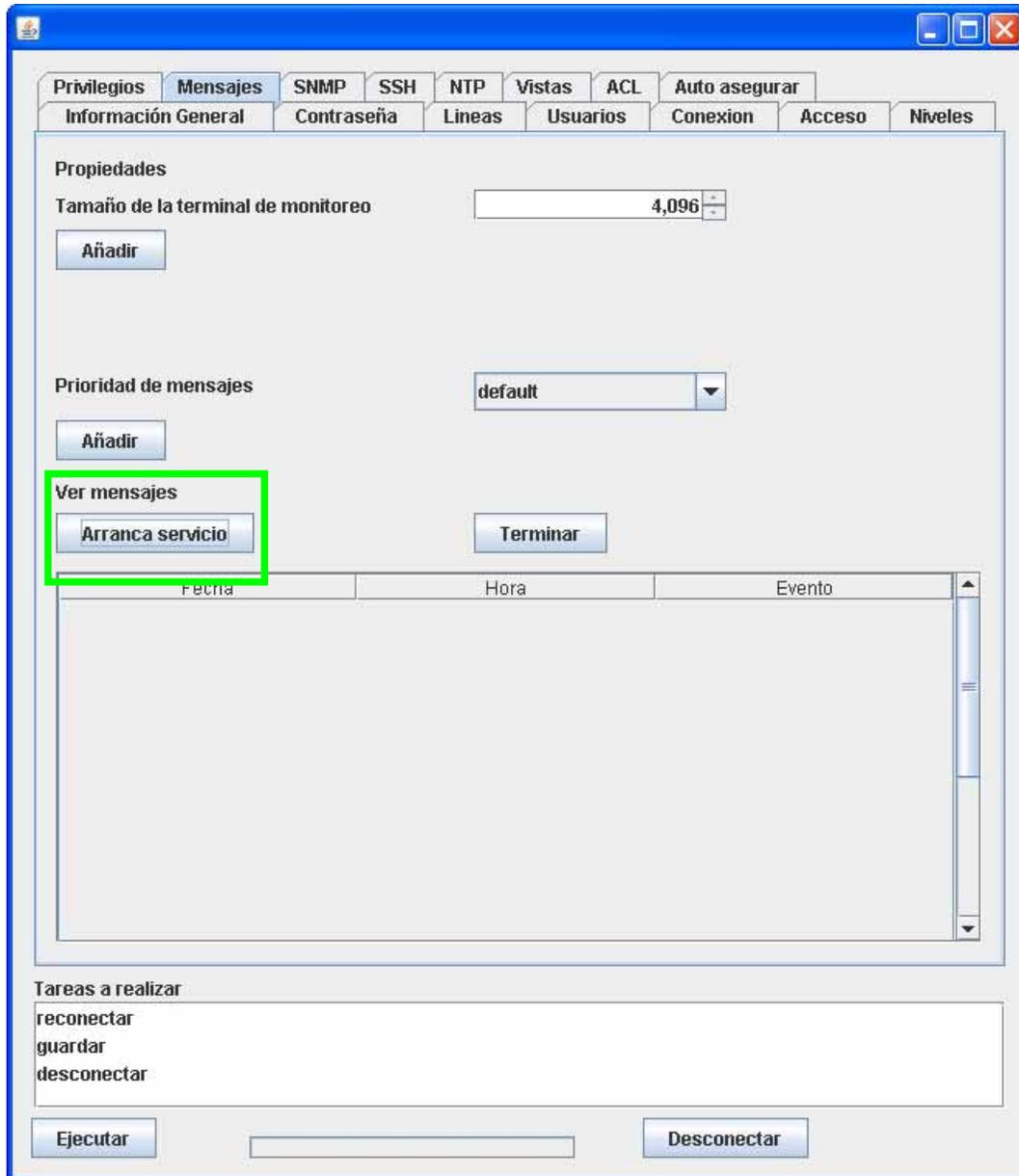
Se comprueban los cambios dentro de la configuración del Router (ruteador).

```
Dynamips(0): R1, Console port
interface FastEthernet0
 ip address 10.1.1.1 255.255.255.0
 speed auto
!
interface Serial0
 no ip address
 shutdown
!
interface Serial1
 no ip address
 shutdown
!
ip forward-protocol nd
!
no ip http server
no ip http secure-server
!
access-list 10 permit 10.1.1.2
access-list 10 permit 10.1.1.3
access-list 10 permit 10.1.1.4
access-list 10 deny 10.1.1.5
access-list 10 deny any log
snmp-server view vista iso.* included
snmp-server view vista system included
snmp-server view vista cisco included
snmp-server community lectura view vista RO
snmp-server community escritura view vista RW
snmp-server enable traps tty
!
control-plane
!
privilege exec all level 5 ping
privilege exec level 2 auto
privilege exec level 4 debug
!
line con 0
 exec-timeout 2 2
line aux 0
 exec-timeout 4 4
--More-- █
```

Prueba 013: mensajes  
Punto de partida.



Presionamos el botón de arrancar.



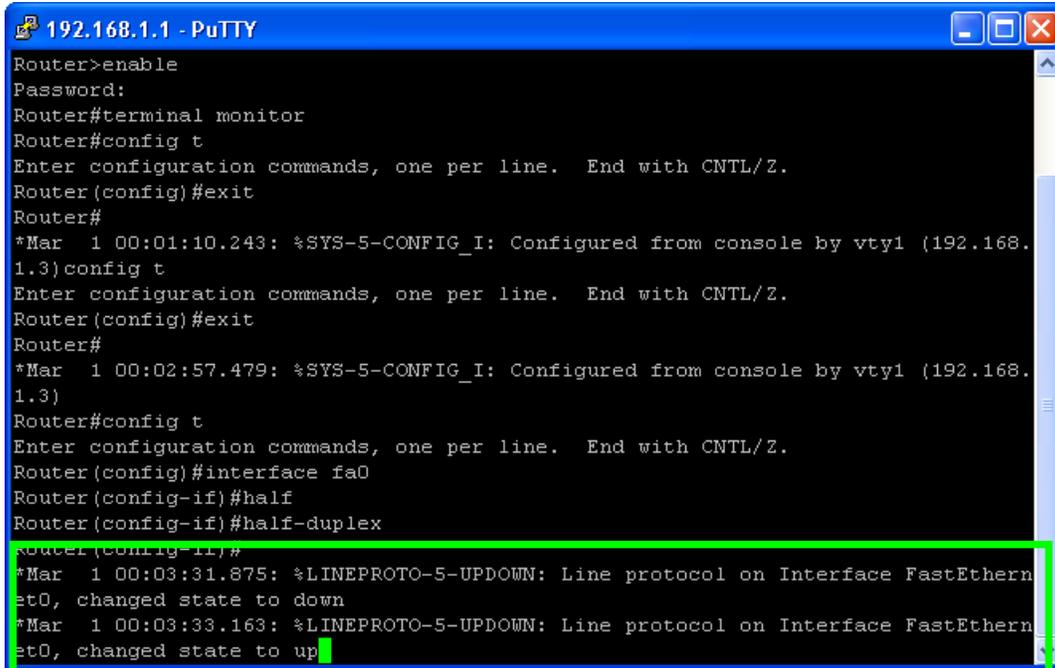
Accedemos al router.

```
192.168.1.1 - PuTTY
User Access Verification
Password:
Router>enable
Password:
Router#terminal monitor
Router#config t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#exit
Router#
*Mar  1 00:01:10.243: %SYS-5-CONFIG_I: Configured from console by vty1 (192.168.1.3)
```

Checamos que en el router la interface FastEthernet 0 este en modo full-duplex.

```
192.168.1.1 - PuTTY
no mmi auto-configure
no mmi pvc
mmi snmp-timeout 180
no aaa new-model
ip subnet-zero
ip cef
!
!
!
no ftp-server write-enable
!
!
!
interface FastEthernet0
 ip address 192.168.1.1 255.255.255.0
 speed auto
 full-duplex
!
ip classless
no ip http server
!
privilege exec all level 8 ping
--More--
```

Cambiamos el modo full-duplex a half-duplex y dos mensajes deben presentarse.



```
192.168.1.1 - PuTTY
Router>enable
Password:
Router#terminal monitor
Router#config t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#exit
Router#
*Mar  1 00:01:10.243: %SYS-5-CONFIG_I: Configured from console by vty1 (192.168.1.3)
Router(config)#exit
Router#
*Mar  1 00:02:57.479: %SYS-5-CONFIG_I: Configured from console by vty1 (192.168.1.3)
Router#config t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#interface fa0
Router(config-if)#half
Router(config-if)#half-duplex
Router(config-if)#
*Mar  1 00:03:31.875: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0, changed state to down
*Mar  1 00:03:33.163: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0, changed state to up
```

Los mismos mensajes se deben de presentar en la aplicación.

Privilegios Mensajes SNMP SSH NTP Vistas ACL Auto asegurar

Información General Contraseña Líneas Usuarios Conexión Acceso Niveles

**Propiedades**

Tamaño de la terminal de monitoreo

Prioridad de mensajes

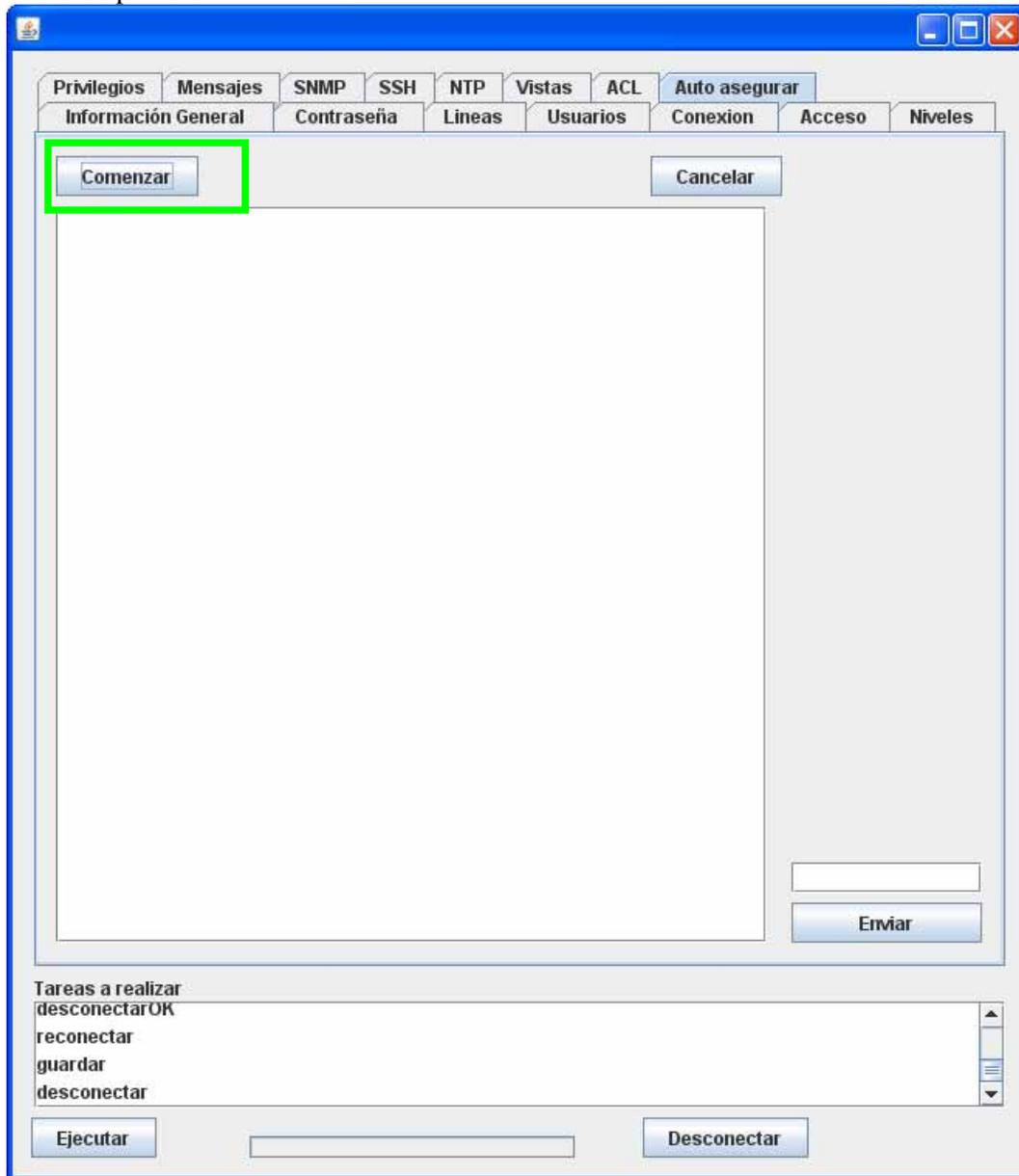
Ver mensajes

| Fecha | H... | Evento   |
|-------|------|--|
| Mar 1 | 0... | %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0, changed state to up   |
| Mar 1 | 0... | %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0, changed state to down |
| Mar 1 | 0... | %SYS-5-CONFIG_I: Configured from console by vty1 (192.168.1.3)                       |

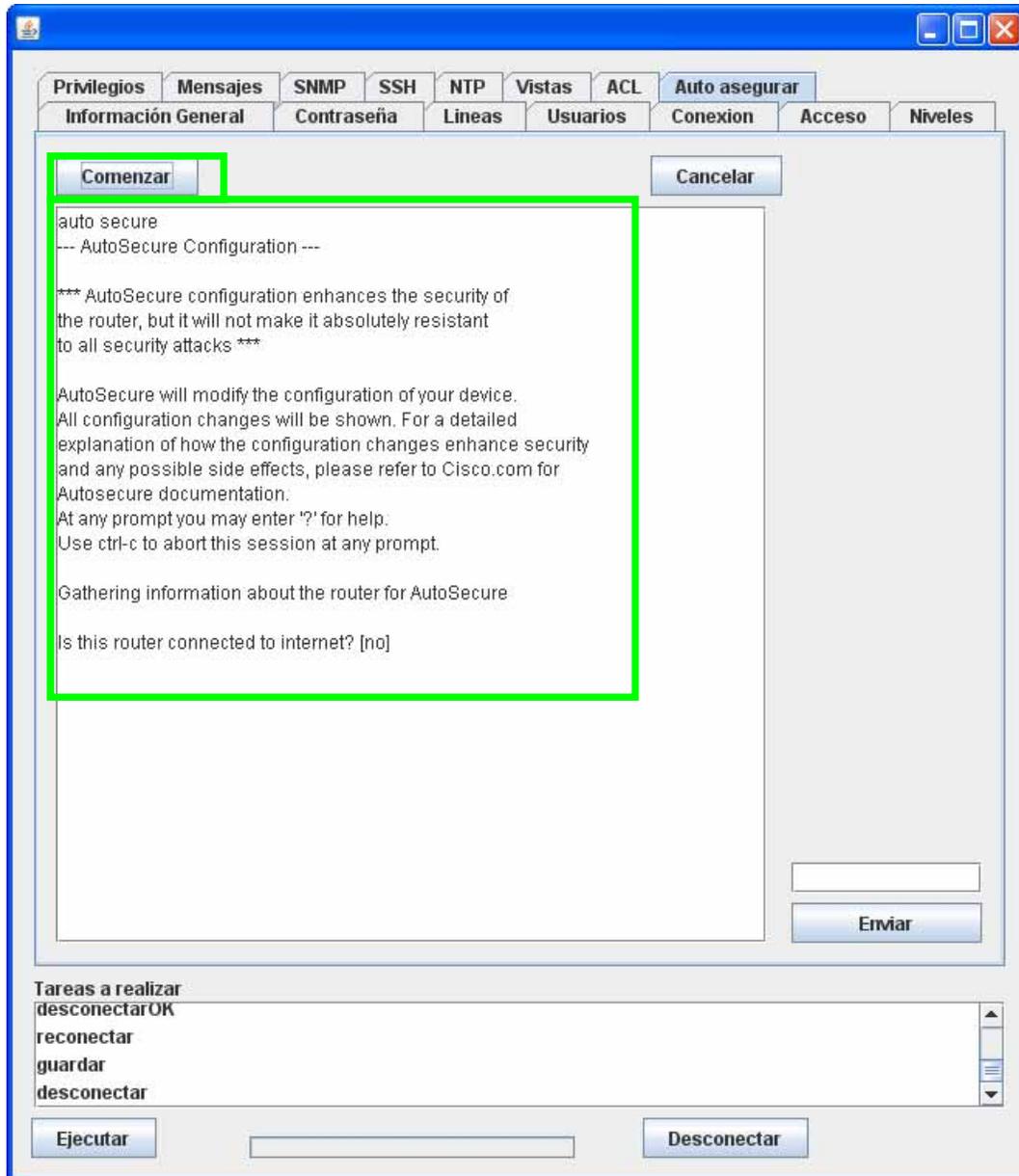
**Tareas a realizar**

reconectar  
guardar  
desconectar

Prueba 014: auto asegurar  
Punto de partida.



Se presiona el botón de comenzar.



Se confirma que el mismo texto y pregunta aparece en el Router (ruteador).

```
Dynamips(0): R1, Console port

Router con0 is now available

Press RETURN to get started.

Router>enable
Password:
Router#auto secure
      --- AutoSecure Configuration ---

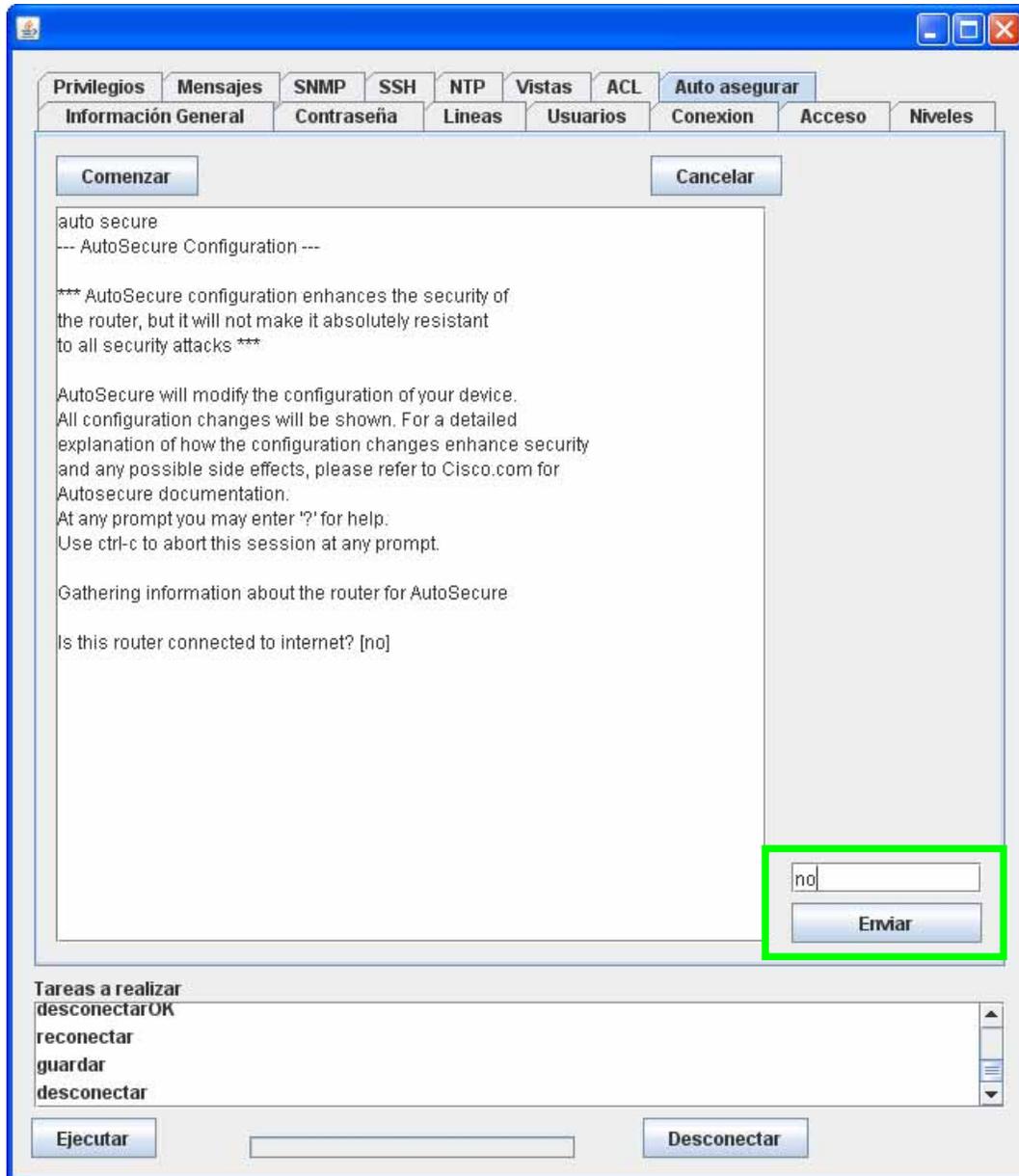
*** AutoSecure configuration enhances the security of
the router, but it will not make it absolutely resistant
to all security attacks ***

AutoSecure will modify the configuration of your device.
All configuration changes will be shown. For a detailed
explanation of how the configuration changes enhance security
and any possible side effects, please refer to Cisco.com for
Autosecure documentation.
At any prompt you may enter '?' for help.
Use ctrl-c to abort this session at any prompt.

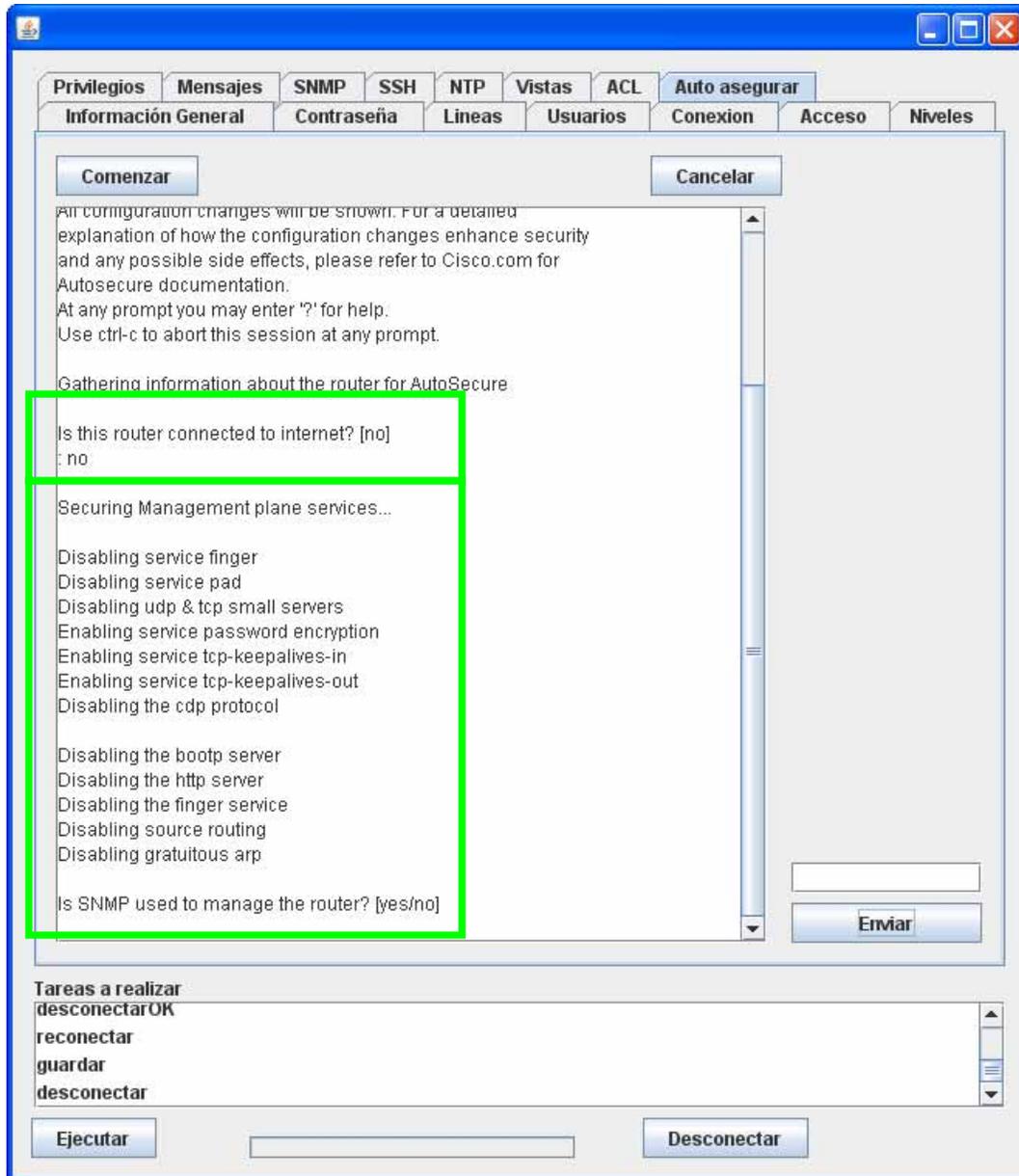
Gathering information about the router for AutoSecure

Is this router connected to internet? [no]: █
```

Se proporciona la respuesta y se presiona el botón de enviar.



La respuesta se manda al Router (ruteador) y regresa a la siguiente parte del texto.



Se confirma el mismo texto dentro del Router (ruteador).

Dynamips(0): R1, Console port

```
Router>enable
Password:
Router#auto secure
      --- AutoSecure Configuration ---

*** AutoSecure configuration enhances the security of
the router, but it will not make it absolutely resistant
to all security attacks ***

AutoSecure will modify the configuration of your device.
All configuration changes will be shown. For a detailed
explanation of how the configuration changes enhance security
and any possible side effects, please refer to Cisco.com for
AutoSecure documentation.
At any prompt you may enter '?' for help.
Use ctrl-c to abort this session at any prompt.

Gathering information about the router for AutoSecure

Is this router connected to internet? [no]: no

Securing Management plane services...

Disabling service finger
Disabling service pad
Disabling udp & tcp small servers
Enabling service password encryption
Enabling service tcp-keepalives-in
Enabling service tcp-keepalives-out
Disabling the cdp protocol

Disabling the bootp server
Disabling the http server
Disabling the finger service
Disabling source routing
Disabling gratuitous arp

Is SNMP used to manage the router? [yes/no]: █
```

Prueba 015: crear lista de acceso.

Punto de partida.

The screenshot shows a network configuration window with the following elements:

- Navigation tabs: Privilegios, Mensajes, SNMP, SSH, NTP, Vistas, **ACL**, Auto asegurar.
- Sub-tabs: Información General, Contraseña, Líneas, Usuarios, Conexión, Acceso, Niveles.
- ACL Configuration: **ACL** section with radio buttons for  Permit and  Deny. Fields for "Número de la lista" (value: 1) and "Host a integrar en la lista".
- ACL List Table (highlighted with a green border):

| Lista | Estado | Anfitrión |
|-------|--------|-----------|
| 10    | deny   | 10.1.1.5  |
| 10    | deny   | any log   |
| 10    | permit | 10.1.1.2  |
| 10    | permit | 10.1.1.3  |
| 10    | permit | 10.1.1.4  |
- Buttons: "Añadir ACL" and "Eliminar".
- Tareas a realizar: reconnectar, guardar, desconectar.
- Buttons: "Ejecutar" and "Desconectar".

Se confirma la misma información dentro del Router (ruteador).

```
Dynamips(0): R1, Console port
interface FastEthernet0
 ip address 10.1.1.1 255.255.255.0
 speed auto
 full-duplex
!
interface Serial0
 no ip address
 shutdown
!
interface Serial1
 no ip address
 shutdown
!
ip forward-protocol nd
!
no ip http server
no ip http secure server
!
access-list 10 permit 10.1.1.2
access-list 10 permit 10.1.1.3
access-list 10 permit 10.1.1.4
access-list 10 deny 10.1.1.5
access-list 10 deny any log
snmp-server view vista iso.* included
snmp-server view vista system included
snmp-server view vista cisco included
snmp-server community lectura view vista RO
snmp-server community escritura view vista RW
snmp-server enable traps tty
!
control-plane
!
privilege exec all level 5 ping
privilege exec level 2 auto
privilege exec level 4 debug
!
line con 0
 exec-timeout 2 2
line aux 0
--More--
```

Selecciona un número de lista de acceso.

Privilegios Mensajes SNMP SSH NTP Vistas ACL Auto asegurar

Información General Contraseña Líneas Usuarios Conexión Acceso Niveles

ACL  Permit  Deny

Número de la lista  Host a integrar en la lista

| Lista | Estado | Anfitrión |
|-------|--------|-----------|
| 10    | deny   | 10.1.1.5  |
| 10    | deny   | any log   |
| 10    | permit | 10.1.1.2  |
| 10    | permit | 10.1.1.3  |
| 10    | permit | 10.1.1.4  |

Tareas a realizar

reconectar  
guardar  
desconectar

Se selecciona un estado y el anfitrión.

Privilegios Mensajes SNMP SSH NTP Vistas ACL Auto asegurar

Información General Contraseña Líneas Usuarios Conexión Acceso Niveles

ACL

Permit

Deny

Número de la lista 11 Host a integrar en la lista 128.1.1.1

| Lista | Estado | Acción   |
|-------|--------|----------|
| 10    | deny   | 10.1.1.5 |
| 10    | deny   | any log  |
| 10    | permit | 10.1.1.2 |
| 10    | permit | 10.1.1.3 |
| 10    | permit | 10.1.1.4 |

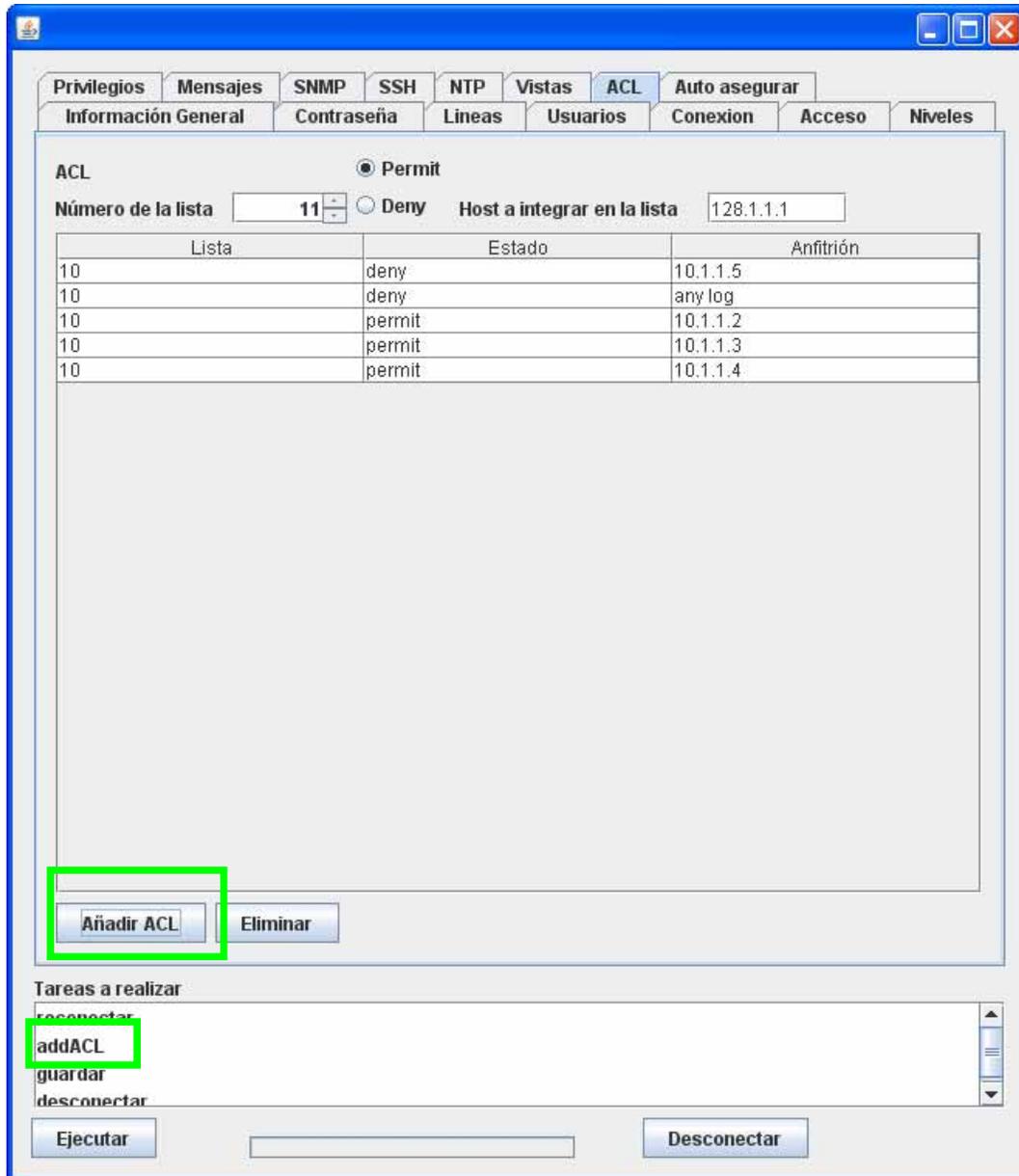
Añadir ACL Eliminar

Tareas a realizar

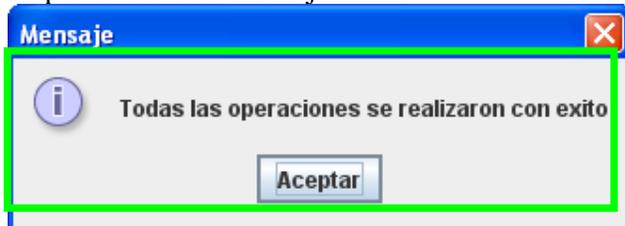
reconectar  
guardar  
desconectar

Ejecutar Desconectar

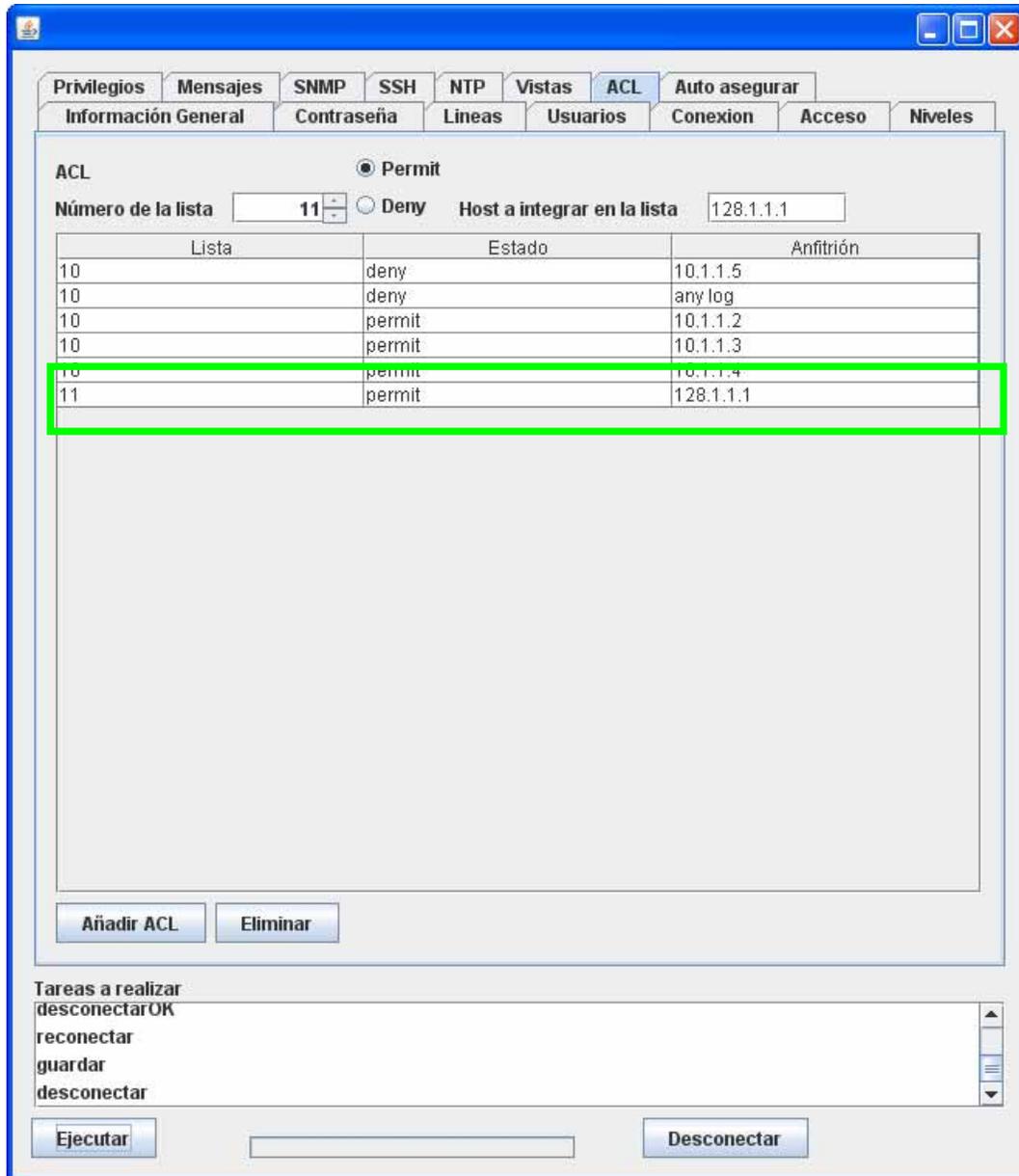
Se presiona el botón de añadir.



Se presiona el botón de ejecutar.



Se aprecian los cambios en la interfaz.



Se comprueban los datos dentro de la configuración del Router (ruteador).

```
Dynamips(0): R1, Console port
interface FastEthernet0
 ip address 10.1.1.1 255.255.255.0
 speed auto
 full-duplex
!
interface Serial0
 no ip address
 shutdown
!
interface Serial1
 no ip address
 shutdown
!
ip forward-protocol nd
!
no ip http server
no ip http secure-server
!
access-list 10 permit 10.1.1.2
access-list 10 permit 10.1.1.3
access-list 10 permit 10.1.1.4
access-list 10 deny 10.1.1.5
access-list 10 deny any log
access-list 11 permit 128.1.1.1
snmp-server view vista 128.1.1.1 included
snmp-server view vista system included
snmp-server view vista cisco included
snmp-server community lectura view vista RO
snmp-server community escritura view vista RW
snmp-server enable traps tty
!
control-plane
!
privilege exec all level 5 ping
privilege exec level 2 auto
privilege exec level 4 debug
!
line con 0
 exec-timeout 2 2
--More--
```

Prueba 016: Eliminar lista de acceso.

Punto de partida

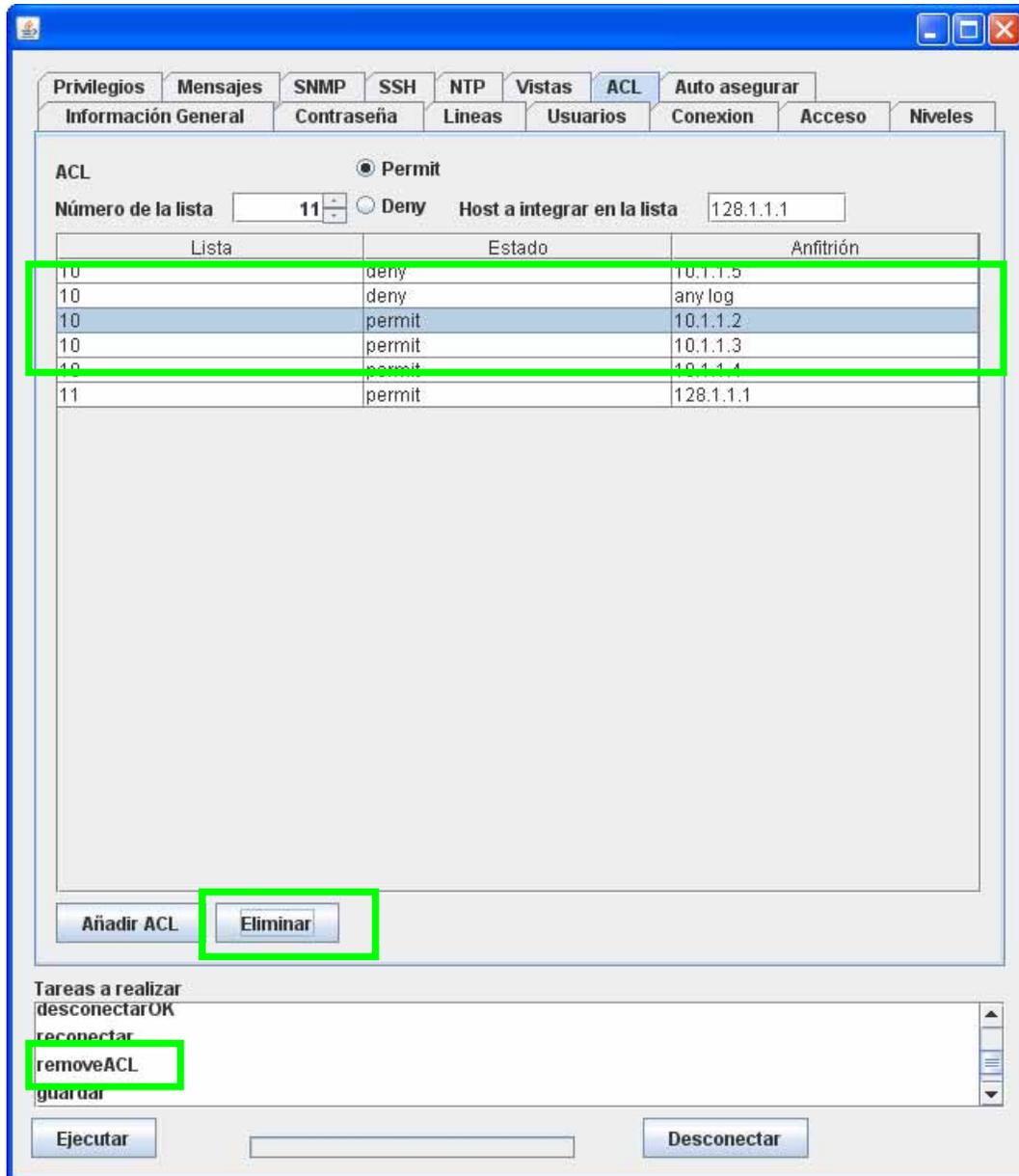
The screenshot shows a network configuration window with the following tabs: Privilegios, Mensajes, SNMP, SSH, NTP, Vistas, ACL, and Auto asegurar. The 'ACL' tab is selected. Below the tabs, there are sub-tabs: Información General, Contraseña, Lineas, Usuarios, Conexion, Acceso, and Niveles. The main area is titled 'ACL' and has radio buttons for 'Permit' (selected) and 'Deny'. Below this, there are input fields for 'Número de la lista' (value: 11) and 'Host a integrar en la lista' (value: 128.1.1.1). A table with three columns (Lista, Estado, Anfitrión) is highlighted with a green border. Below the table are buttons for 'Añadir ACL' and 'Eliminar'. At the bottom, there is a 'Tareas a realizar' section with a list of tasks: desconectarOK, reconectar, guardar, and desconectar. At the very bottom, there are buttons for 'Ejecutar' and 'Desconectar'.

| Lista | Estado | Anfitrión |
|-------|--------|-----------|
| 10    | deny   | 10.1.1.5  |
| 10    | deny   | any log   |
| 10    | permit | 10.1.1.2  |
| 10    | permit | 10.1.1.3  |
| 10    | permit | 10.1.1.4  |
| 11    | permit | 128.1.1.1 |

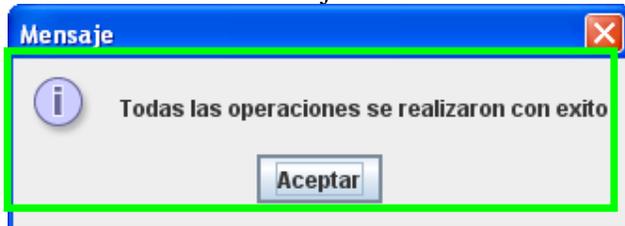
Comprobamos la misma información dentro del Router (ruteador).

```
Dynamips(0): R1, Console port
interface FastEthernet0
 ip address 10.1.1.1 255.255.255.0
 speed auto
 full-duplex
!
interface Serial0
 no ip address
 shutdown
!
interface Serial1
 no ip address
 shutdown
!
ip forward-protocol nd
!
no ip http server
no ip http secure-server
!
access-list 10 permit 10.1.1.2
access-list 10 permit 10.1.1.3
access-list 10 permit 10.1.1.4
access-list 10 deny 10.1.1.5
access-list 10 deny any log
access-list 11 permit 128.1.1.1
!
snmp-server view vista system included
snmp-server view vista cisco included
snmp-server community lectura view vista RO
snmp-server community escritura view vista RW
snmp-server enable traps tty
!
control-plane
!
privilege exec all level 5 ping
privilege exec level 2 auto
privilege exec level 4 debug
!
line con 0
 exec-timeout 2 2
--More--
```

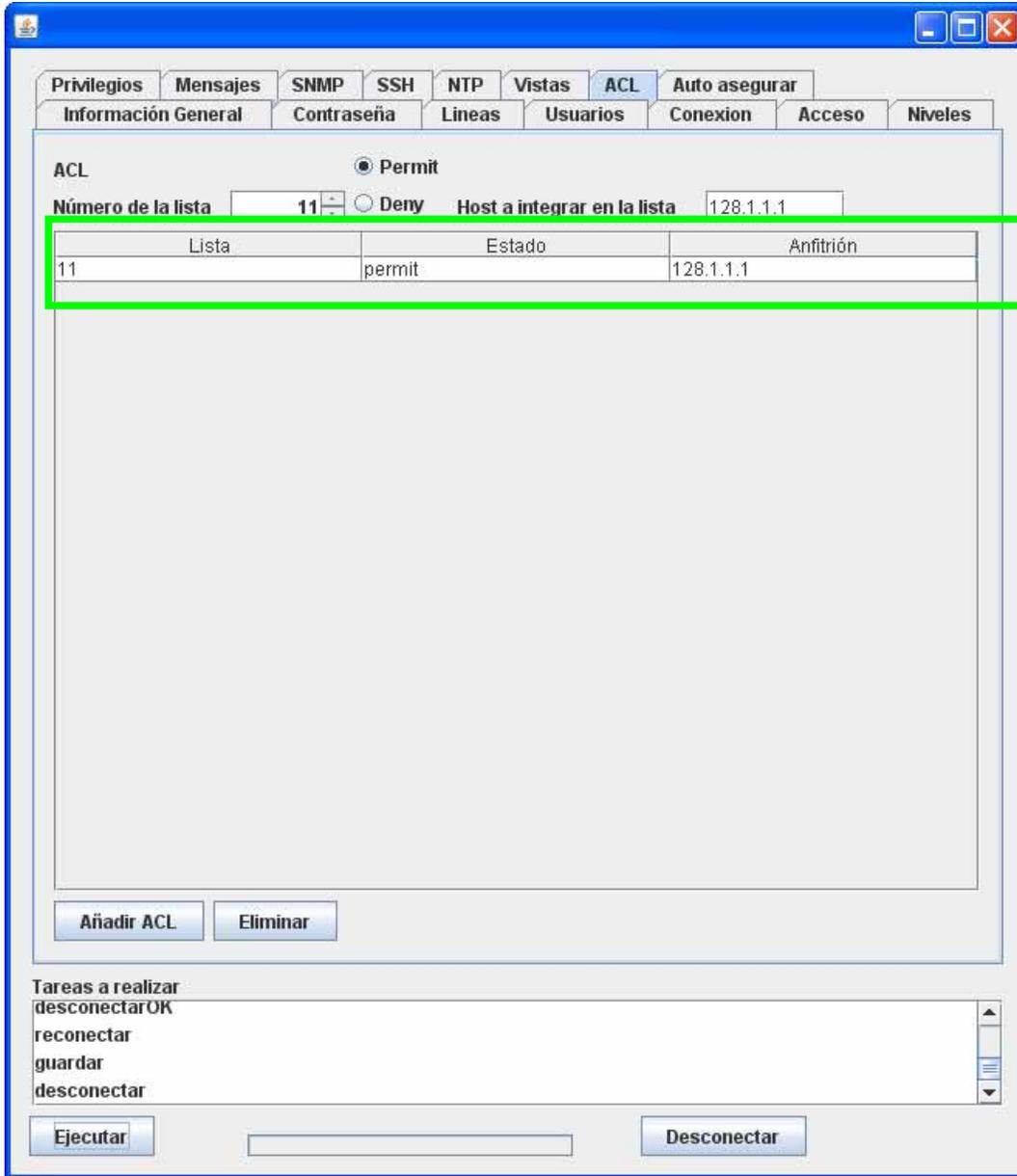
Seleccionamos una lista en la interfaz. Y presionamos el botón de eliminar.



Presionamos el botón de ejecutar.



Observamos los cambios en la interfaz.



Compró los cambios dentro del Router (ruteador).

```
Dynamips(0): R1, Console port
interface FastEthernet0
 ip address 10.1.1.1 255.255.255.0
 speed auto
 full-duplex
!
interface Serial0
 no ip address
 shutdown
!
interface Serial1
 no ip address
 shutdown
!
ip forward-protocol nd
!
no ip http server
no ip http secure server
!
access-list 11 permit 128.1.1.1
snmp-server view vista iso.* included
snmp-server view vista system included
snmp-server view vista cisco included
snmp-server community lectura view vista RO
snmp-server community escritura view vista RW
snmp-server enable traps tty
!
control-plane
!
privilege exec all level 5 ping
privilege exec level 2 auto
privilege exec level 4 debug
!
line con 0
 exec-timeout 2 2
line aux 0
 exec-timeout 4 4
line vty 0
 exec-timeout 5 15
 password linea
--More--
```

Prueba 017: modificar una vista.

Unto de partida.

**Vistas**

Nombre de la vista  MIB a incluir   Included  Excluded

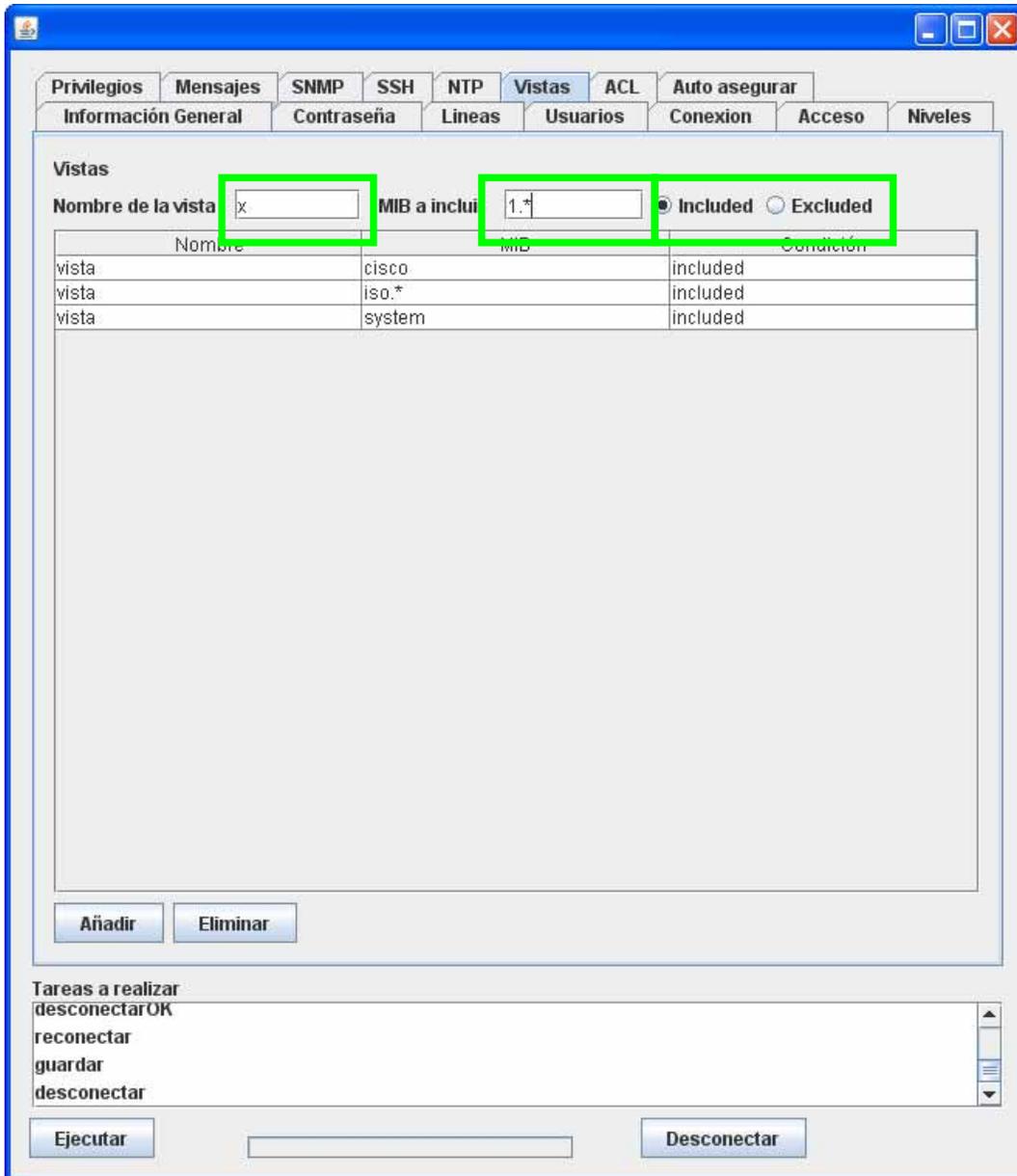
| Nombre | MIB    | Condición |
|--------|--------|-----------|
| vista  | cisco  | included  |
| vista  | iso.*  | included  |
| vista  | system | included  |

Tareas a realizar  
desconectarOK  
reconectar  
guardar  
desconectar

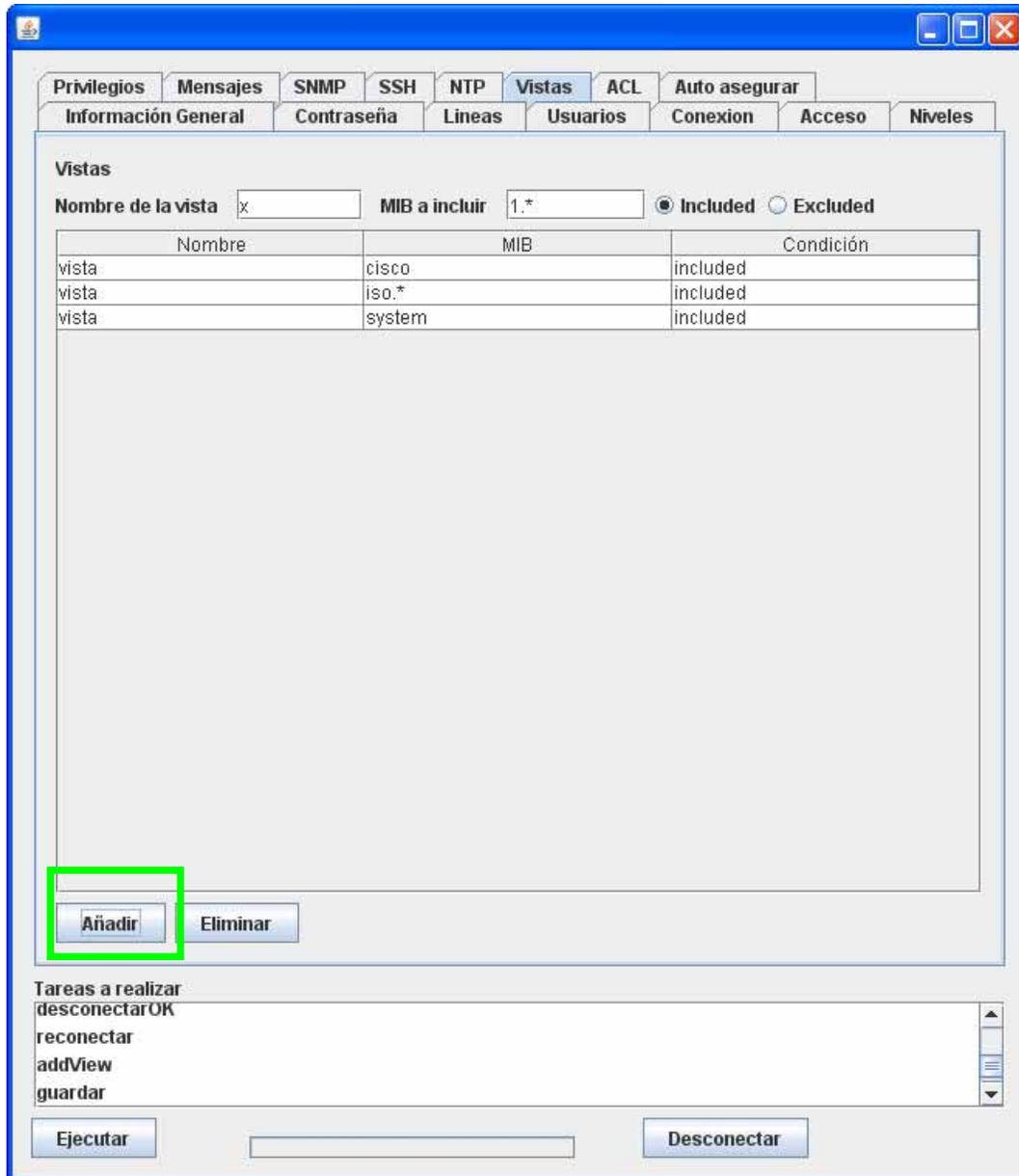
Confirmamos la misma información dentro del Router (ruteador).

```
Dynamips(0): R1, Console port
interface FastEthernet0
 ip address 10.1.1.1 255.255.255.0
 speed auto
 full-duplex
!
interface Serial0
 no ip address
 shutdown
!
interface Serial1
 no ip address
 shutdown
!
ip forward-protocol nd
!
no ip http server
no ip http secure-server
!
access-list 10 permit 10.1.1.1
snmp-server view vista iso.* included
snmp-server view vista system included
snmp-server view vista cisco included
snmp-server community lectura view vista RO
snmp-server community escritura view vista RW
snmp-server enable traps tty
!
control-plane
!
privilege exec all level 5 ping
privilege exec level 2 auto
privilege exec level 4 debug
!
line con 0
 exec-timeout 2 2
line aux 0
 exec-timeout 4 4
line vty 0
 exec-timeout 5 15
 password linea
--More--
```

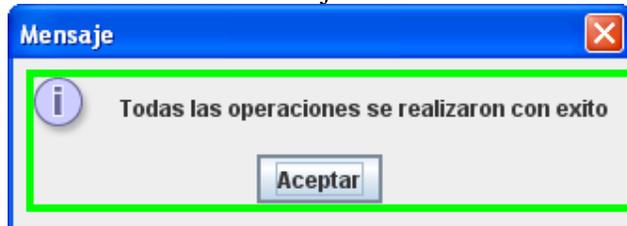
Proporcionamos un hombre de la vista y la mide incluir y seleccionamos entre la opción included y excluded.



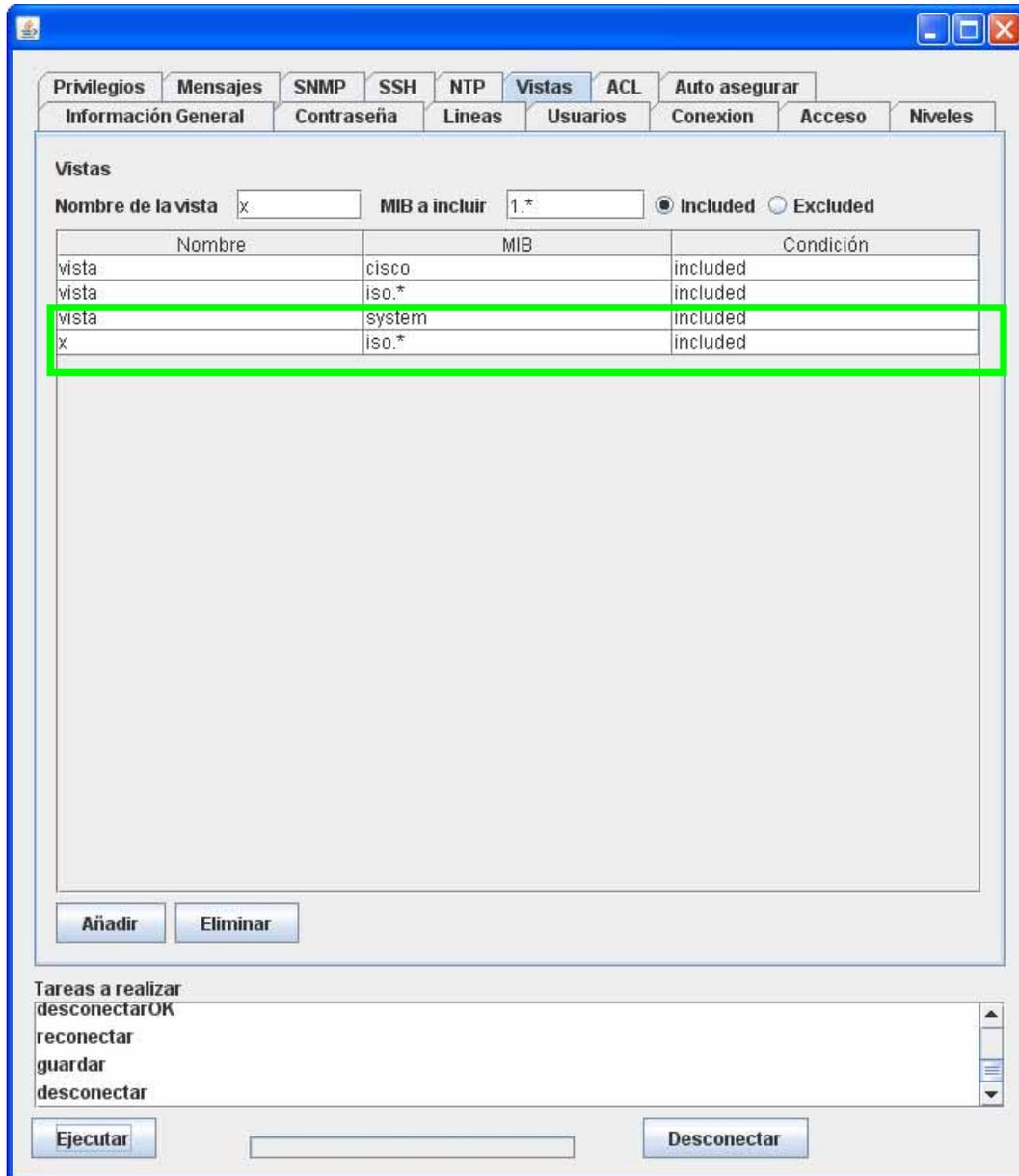
Presionamos el botón de añadir.



Presionamos el botón de ejecutar.



Observamos los cambios en la interfaz.



Comprobamos cambios dentro de la configuración del Router (ruteador).

```
Dynamips(0): R1, Console port
interface FastEthernet0
 ip address 10.1.1.1 255.255.255.0
 speed auto
 full-duplex
!
interface Serial0
 no ip address
 shutdown
!
interface Serial1
 no ip address
 shutdown
!
ip forward-protocol nd
!
no ip http server
no ip http secure-server
!
access-list 10 permit 10.1.1.1
snmp-server view x iso.* included
snmp-server view vista iso.* included
snmp-server view vista system included
snmp-server view vista cisco included
snmp-server community lectura view vista RO
snmp-server community escritura view vista RW
snmp-server enable traps tty
!
control-plane
!
privilege exec all level 5 ping
privilege exec level 2 auto
privilege exec level 4 debug
!
line con 0
 exec-timeout 2 2
line aux 0
 exec-timeout 4 4
line vty 0
 exec-timeout 5 15
--More--
```

Prueba 019: eliminar lista de acceso.

Punto de partida.

**Vistas**

Nombre de la vista: x      MIB a incluir: 1.\*       Included       Excluded

| Nombre | MIB    | Condición |
|--------|--------|-----------|
| vista  | cisco  | included  |
| vista  | iso.*  | included  |
| vista  | system | included  |
| x      | iso.*  | included  |

**Tareas a realizar**

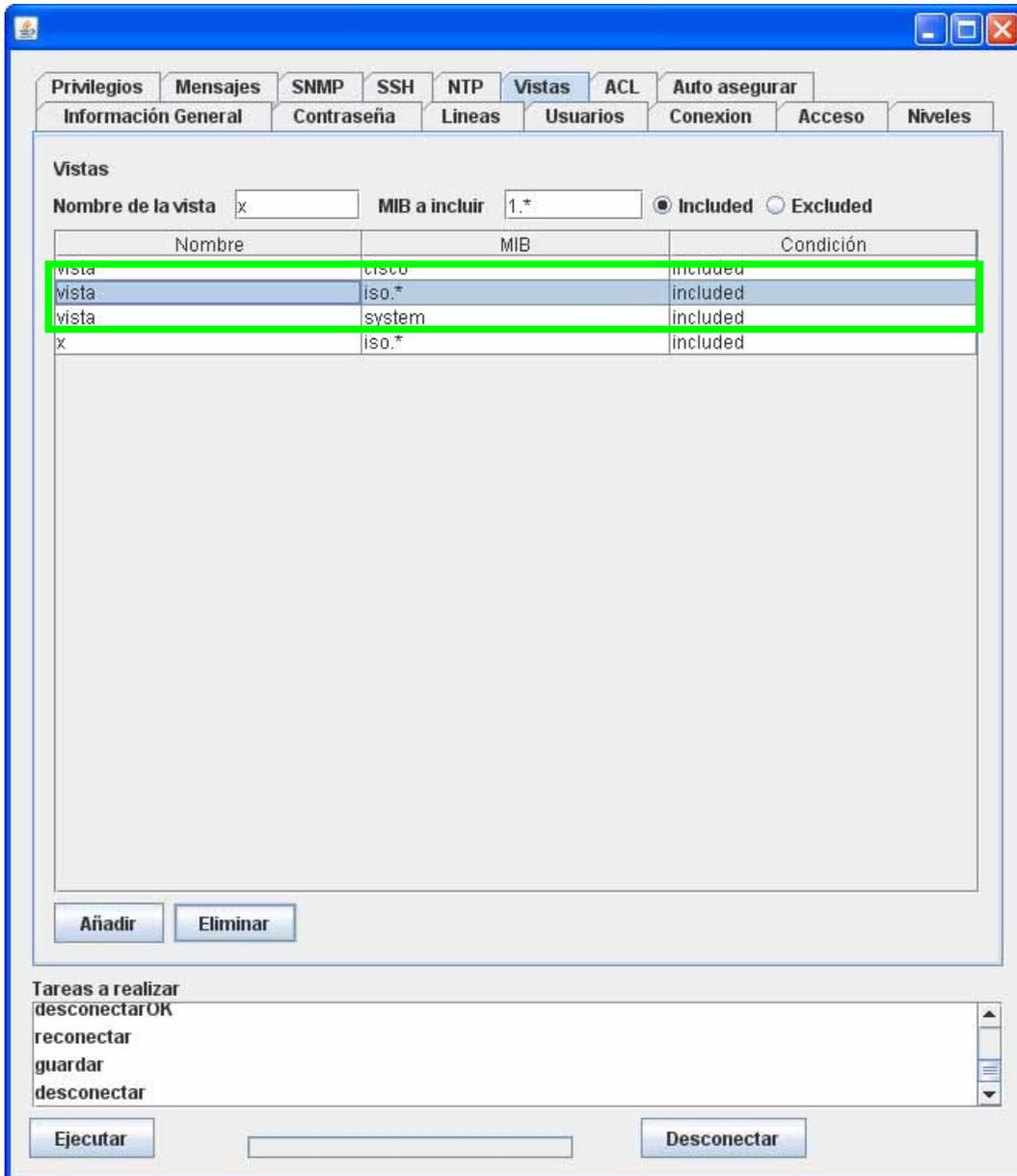
- desconectarOK
- reconectar
- guardar
- desconectar

Ejecutar      Desconectar

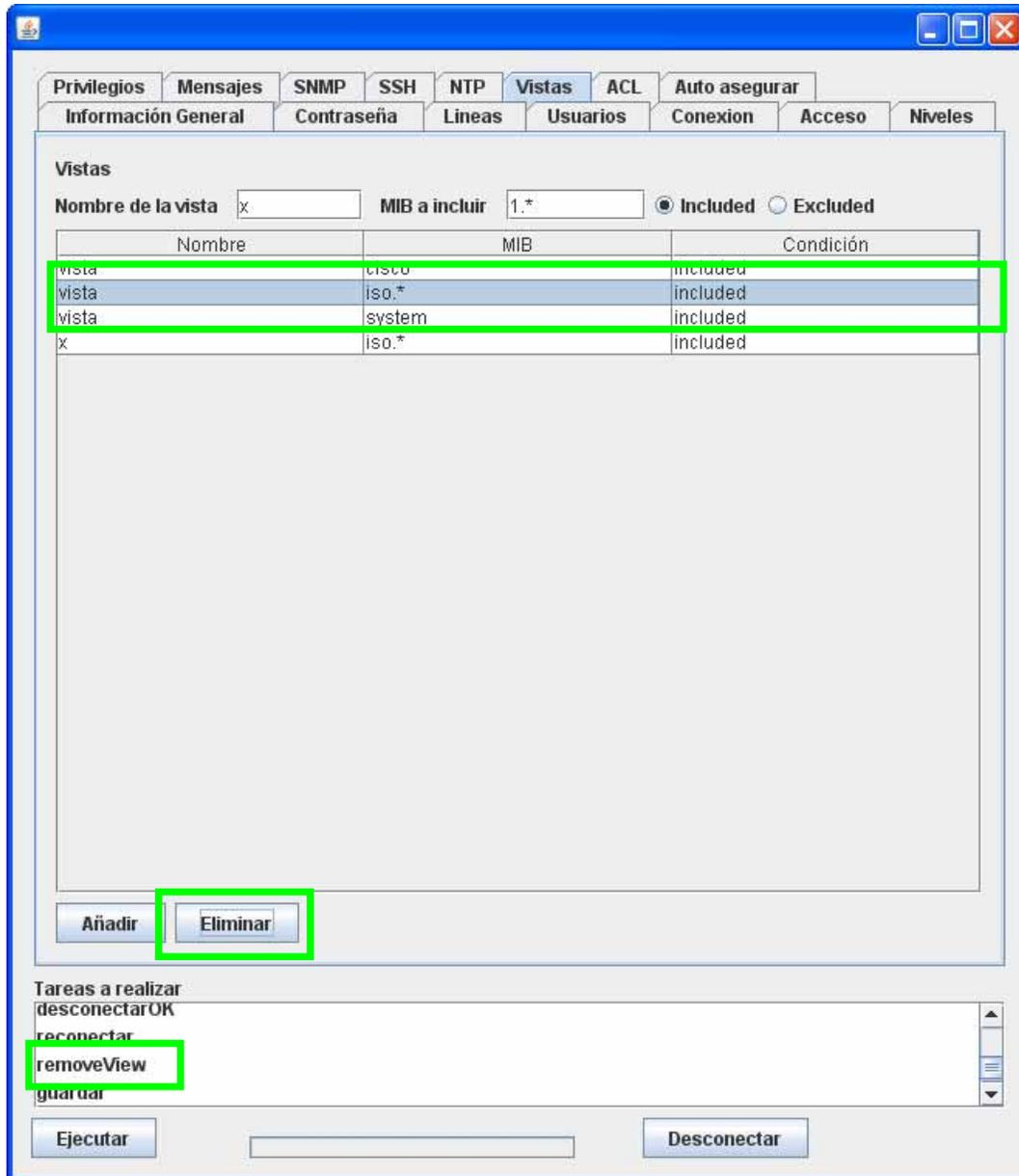
Comprobamos la información dentro de la configuración del Router (ruteador)

```
Dynamips(0): R1, Console port
speed auto
full-duplex
!
interface Serial0
no ip address
shutdown
!
interface Serial1
no ip address
shutdown
!
ip forward-protocol nd
!
no ip http server
no ip http secure-server
!
access-list 11 permit 128.1.1.1
snmp-server view x iso.* included
snmp-server view vista iso.* included
snmp-server view vista system included
snmp-server view vista cisco included
snmp-server community lectura view vista RO
snmp-server community escritura view vista RW
snmp-server enable traps tty
!
control-plane
!
privilege exec all level 5 ping
privilege exec level 2 auto
privilege exec level 4 debug
!
line con 0
exec-timeout 2 2
line aux 0
exec-timeout 4 4
line vty 0
exec-timeout 5 15
password linea
line vty 1
password linea
```

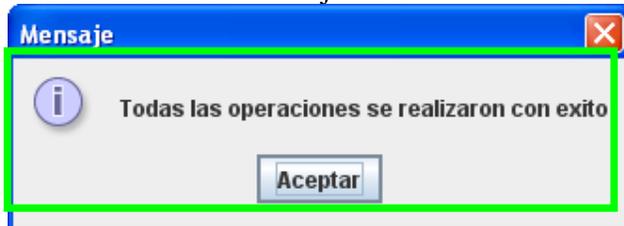
Se selecciona una vista de la tabla de vistas.



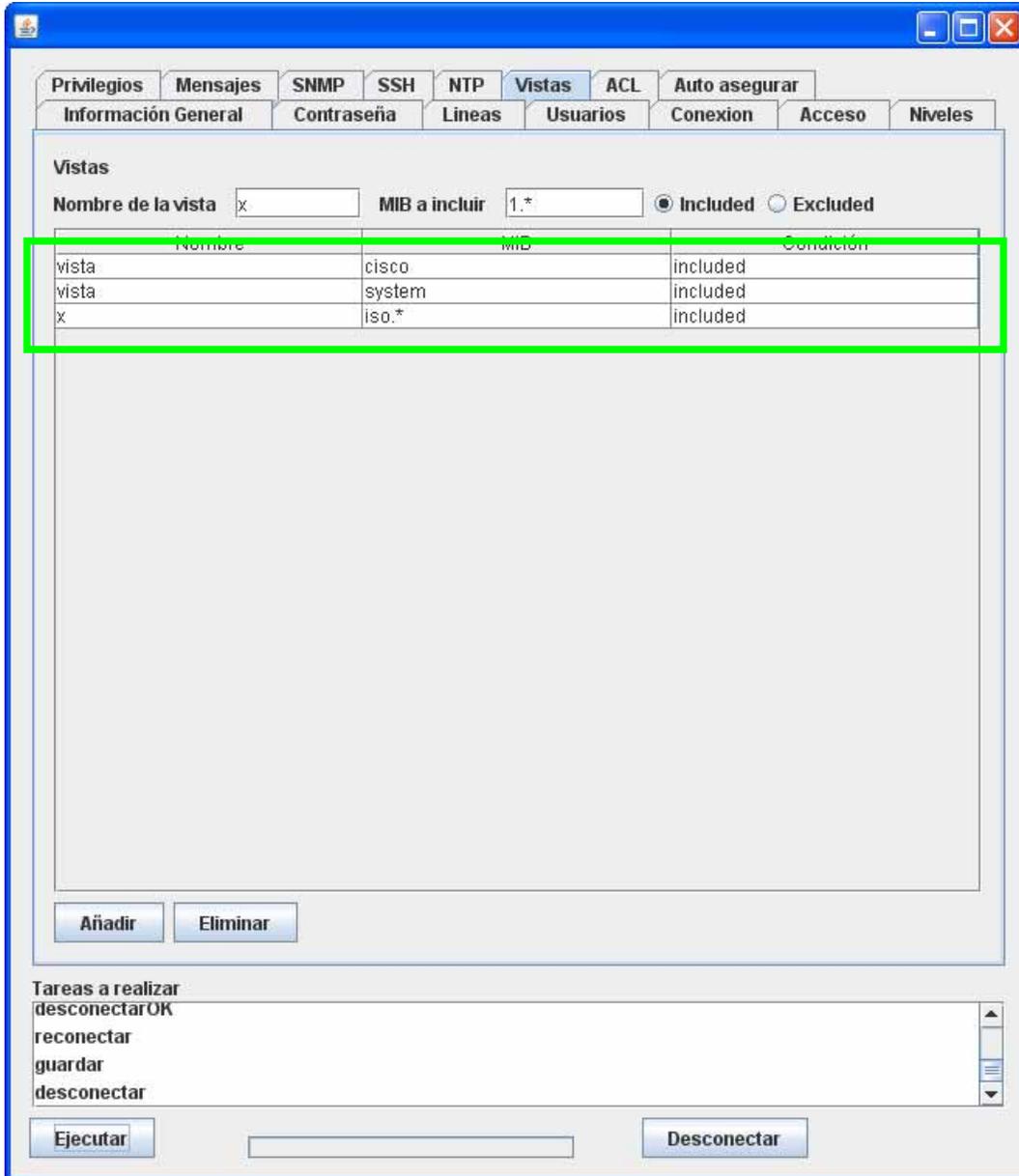
Se presiona el botón de eliminar.



Presionamos el botón de ejecutar.



Observamos los cambios en la interfaz.



Comprobamos los cambios dentro de la configuración del Router (ruteador).

```
Dynamips(0): R1, Console port
interface FastEthernet0
 ip address 10.1.1.1 255.255.255.0
 speed auto
 full-duplex
!
interface Serial0
 no ip address
 shutdown
!
interface Serial1
 no ip address
 shutdown
!
ip forward-protocol nd
!
no ip http server
no ip http secure-server
!
access-list 10 permit 10.1.1.1
snmp-server view x iso.* included
snmp-server view vista system included
snmp-server view vista cisco included
snmp-server community lectura view vista 10
snmp-server community escritura view vista RW
snmp-server enable traps tty
!
control-plane
!
privilege exec all level 5 ping
privilege exec level 2 auto
privilege exec level 4 debug
!
line con 0
 exec-timeout 2 2
line aux 0
 exec-timeout 4 4
line vty 0
 exec-timeout 5 15
 password linea
--More--
```

Prueba 020: establecer un servidor NTP  
Punto de partida

Privilegios Mensajes SNMP SSH **NTP** Vistas ACL Auto asegurar

Información General Contraseña Líneas Usuarios Conexion Acceso Niveles

Tipo de servicio  
NTP

Nombre o ip del servidor  
  Preferida

Añadir

| Tipo de servicio | Nombre o ip | Preferido |
|------------------|-------------|-----------|
| ntp              | 10.1.1.3    | Sí        |
| ntp              | 10.1.1.2    | Sí        |

Eliminar

Tareas a realizar  
desconectarOK  
reconectar  
guardar  
desconectar

Ejecutar  Desconectar

Comprobamos la configuración dentro del Router (ruteador).

```
Dynamips(0): R1, Console port
snmp-server view vista cisco included
snmp-server community lectura view vista RO
snmp-server community escritura view vista RW
snmp-server enable traps tty
!
control-plane
!
privilege exec all level 5 ping
privilege exec level 2 auto
privilege exec level 4 debug
!
line con 0
  exec-timeout 2 2
line aux 0
  exec-timeout 4 4
line vty 0
  exec-timeout 5 15
  password linea
line vty 1
  password linea
line vty 2
  exec-timeout 3 2
  password jojojo
line vty 3
  password hola
line vty 4
  password adios
parser view vista1
secret 5 $1$TVOB$E5HmQ3ZnI4zGeF1S2ch2C.
commands exec include all ping
commands exec include show running-config
commands exec include show
commands exec exclude debug
!
ntp server 10.1.1.3 prefer
ntp server 10.1.1.2 prefer
end

Router#
```

Proporcionamos la IP (protocolo de Internet) del servidor de NTP.

Privilegios Mensajes SNMP SSH NTP Vistas ACL Auto asegurar

Información General Contraseña Líneas Usuarios Conexión Acceso Niveles

Tipo de servicio  
ntp

Nombre o ip del servidor  
192.168.1.1  Preferida

Añadir

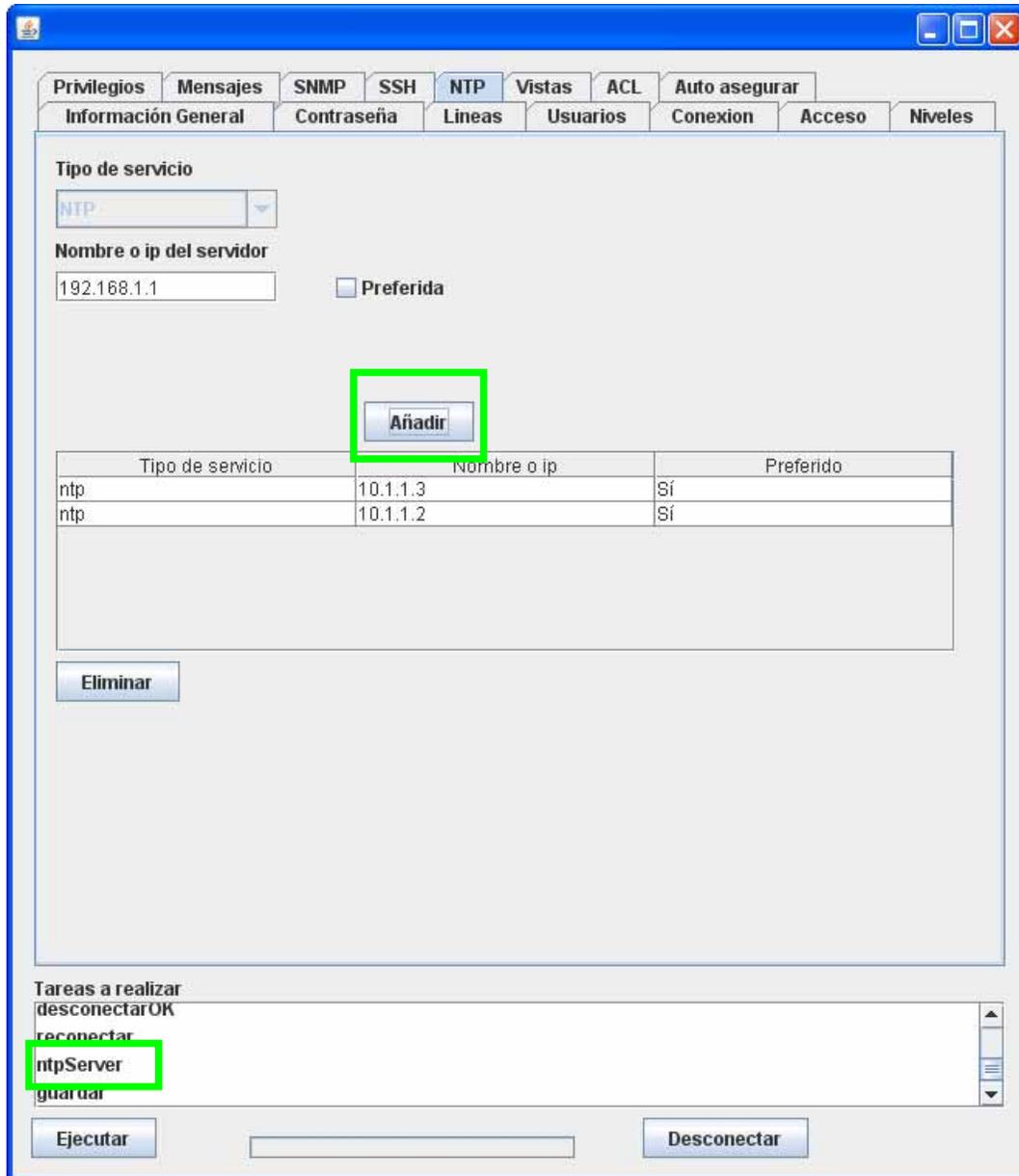
| Tipo de servicio | Nombre o ip | Preferido |
|------------------|-------------|-----------|
| ntp              | 10.1.1.3    | Sí        |
| ntp              | 10.1.1.2    | Sí        |

Eliminar

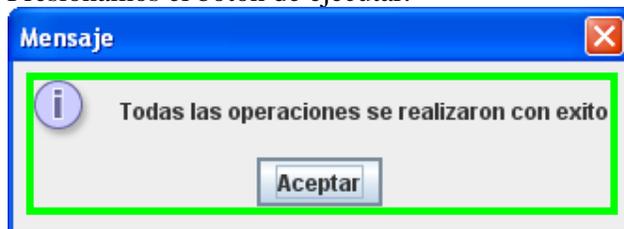
Tareas a realizar  
desconectarOK  
reconectar  
guardar  
desconectar

Ejecutar  Desconectar

Presionamos el botón de añadir.



Presionamos el botón de ejecutar.



Observamos los cambios en la interfaz.

Privilegios Mensajes SNMP SSH **NTP** Vistas ACL Auto asegurar

Información General Contraseña Líneas Usuarios Conexion Acceso Niveles

Tipo de servicio

Nombre o ip del servidor  
  Preferida

| Tipo de servicio | Nombre o ip | Preferido |
|------------------|-------------|-----------|
| ntp              | 192.168.1.1 | No        |
| ntp              | 10.1.1.3    | Sí        |
| ntp              | 10.1.1.2    | Sí        |

Tareas a realizar  
 desconectarOK  
 reconectar  
 guardar  
 desconectar

Comparamos los cambios en la configuración del Router (ruteador).

```
Dynamips(0): R1, Console port
snmp-server community lectura view vista RO
snmp-server community escritura view vista RW
snmp-server enable traps tty
!
control-plane
!
privilege exec all level 5 ping
privilege exec level 2 auto
privilege exec level 4 debug
!
line con 0
  exec-timeout 2 2
line aux 0
  exec-timeout 4 4
line vty 0
  exec-timeout 5 15
  password linea
line vty 1
  password linea
line vty 2
  exec-timeout 3 2
  password jojojo
line vty 3
  password hola
line vty 4
  password adios
parser view vista1
  secret 5 $1$TVOB$E5HmQ3ZnI4zGeF1S2ch2C.
  commands exec include all ping
  commands exec include show running-config
  commands exec include show
  commands exec exclude debug
!
ntp server 192.168.1.1
ntp server 10.1.1.3 prefer
ntp server 10.1.1.2 prefer
end

Router#
```

Prueba 021: Elimina un servidor NTP  
Punto de partida.

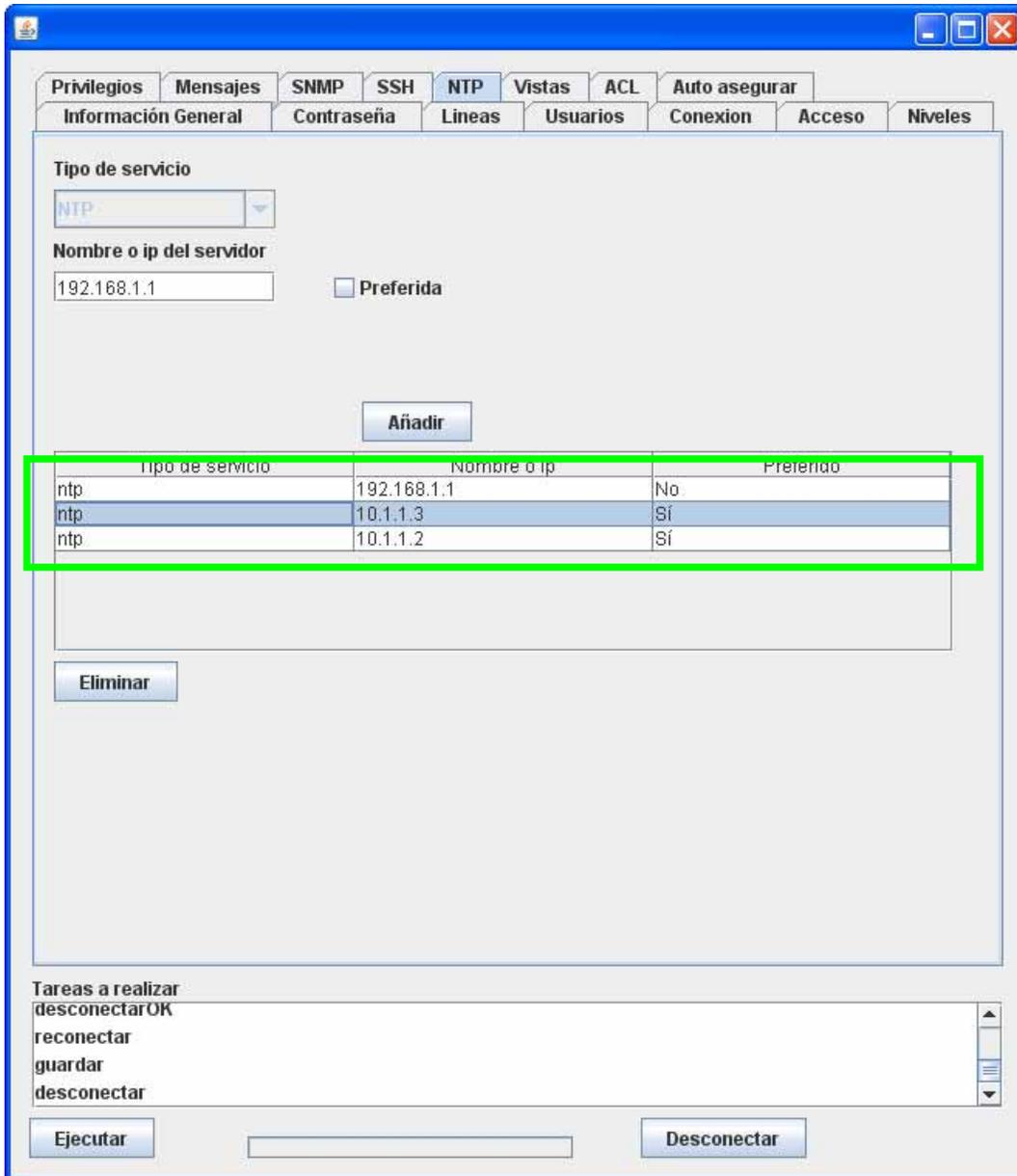
The screenshot shows a web-based configuration interface for a network device. The 'NTP' tab is selected. The 'Información General' sub-tab is active. The 'Tipo de servicio' dropdown is set to 'NTP'. The 'Nombre o ip del servidor' field contains '192.168.1.1' and the 'Preferida' checkbox is unchecked. An 'Añadir' button is visible. Below is a table with three rows of NTP server configurations. The first row is highlighted with a green border. Below the table is an 'Eliminar' button. At the bottom, there is a 'Tareas a realizar' section with a list of actions: 'desconectarOK', 'reconectar', 'guardar', and 'desconectar'. 'Ejecutar' and 'Desconectar' buttons are also present.

| Tipo de servicio | Nombre o ip | Preferido |
|------------------|-------------|-----------|
| ntp              | 192.168.1.1 | No        |
| ntp              | 10.1.1.3    | Sí        |
| ntp              | 10.1.1.2    | Sí        |

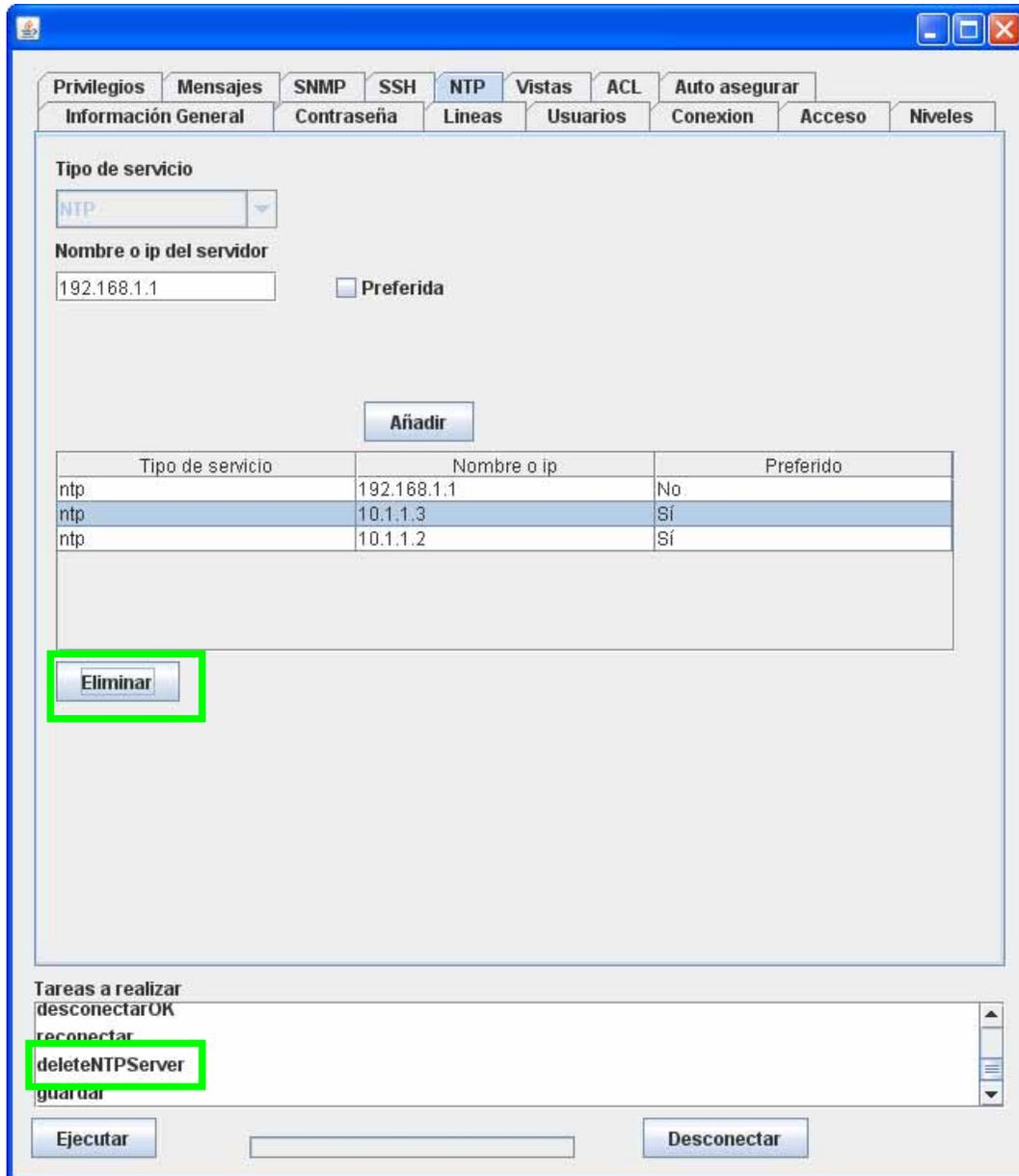
Comprobamos la información dentro de la configuración del Router (ruteador)

```
Dynamips(0): R1, Console port
snmp-server community lectura view vista RO
snmp-server community escritura view vista RW
snmp-server enable traps tty
!
control-plane
!
privilege exec all level 5 ping
privilege exec level 2 auto
privilege exec level 4 debug
!
line con 0
  exec-timeout 2 2
line aux 0
  exec-timeout 4 4
line vty 0
  exec-timeout 5 15
  password linea
line vty 1
  password linea
line vty 2
  exec-timeout 3 2
  password jojojo
line vty 3
  password hola
line vty 4
  password adios
parser view vista1
secret 5 $1$TVOB$E5HmQ3ZnI4zGeF1S2ch2C.
commands exec include all ping
commands exec include show running-config
commands exec include show
commands exec exclude debug
!
ntp server 192.168.1.1
ntp server 10.1.1.3 prefer
ntp server 10.1.1.2 prefer
end
Router#
```

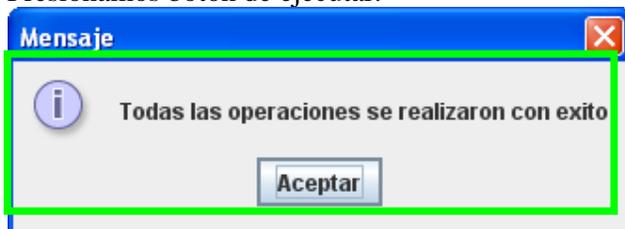
Seleccionamos un servidor de la tabla de servidores.



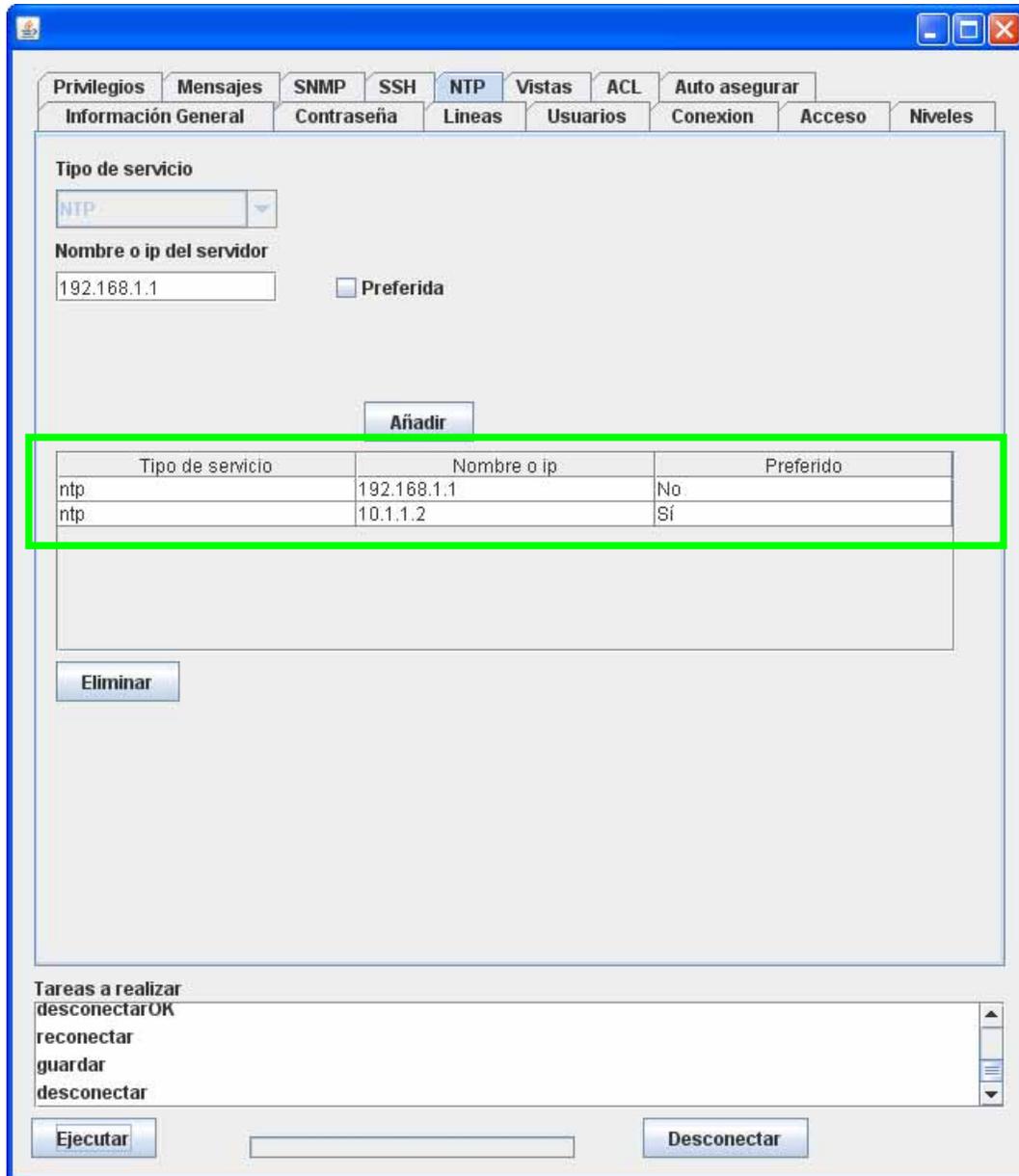
Presionamos el botón de eliminar.



Presionamos botón de ejecutar.



Observamos los cambios en la interfaz.



Comprobamos los cambios en la configuración del Router (ruteador).

```
Dynamips(0): R1, Console port
snmp-server view vista cisco included
snmp-server community lectura view vista RO
snmp-server community escritura view vista RW
snmp-server enable traps tty
!
control-plane
!
privilege exec all level 5 ping
privilege exec level 2 auto
privilege exec level 4 debug
!
line con 0
  exec-timeout 2 2
line aux 0
  exec-timeout 4 4
line vty 0
  exec-timeout 5 15
  password linea
line vty 1
  password linea
line vty 2
  exec-timeout 3 2
  password jojojo
line vty 3
  password hola
line vty 4
  password adios
parser view vista1
secret 5 $1$TVOB$E5HmQ3ZnI4zGeF1S2ch2C.
commands exec include all ping
commands exec include show running-config
commands exec include show
commands exec exclude debug
!
ntp server 192.168.1.1
ntp server 10.1.1.2 prefer
end
Router#
```

Prueba: 022: configurar Servidor SSH  
Punto de partida.

The screenshot shows a web-based configuration interface for SSH. At the top, there are several tabs: Privilegios, Mensajes, SNMP, SSH (selected), NTP, Vistas, ACL, and Auto asegurar. Below these are sub-tabs: Información General, Contraseña, Lineas, Usuarios, Conexion, Acceso, and Niveles. The main content area is titled 'Llave RSA' and contains a 'Mostrar llave actual' button, a large empty text box, a checkbox for 'Generar nueva llave.', a 'Tamaño de módulo' spinner set to 360, and input fields for 'Anfitrión' and 'Dominio'. An 'Añadir' button is located below these fields. At the bottom, there is a 'Tareas a realizar' section with a list of actions: desconectarOK, reconectar, guardar, and desconectar. Below this list are 'Ejecutar' and 'Desconectar' buttons, with an empty input field between them.

Presionamos el botón de mostrar se debe actuar

The screenshot shows a web-based configuration interface for SSH keys. At the top, there are several tabs: Privilegios, Mensajes, SNMP, SSH (selected), NTP, Vistas, ACL, and Auto asegurar. Below these are sub-tabs: Información General, Contraseña, Lineas, Usuarios, Conexion, Acceso, and Niveles. The main content area is titled "Llave RSA" and contains a button "Mostrar llave actual" (highlighted with a green box). Below this button is a text area containing "No hay llave actualmente" (also highlighted with a green box). There is a checkbox labeled "Generar nueva llave." which is currently unchecked. Below the checkbox is a "Tamaño de módulo" field with a value of "360" and a spinner control. Underneath are two text input fields labeled "Anfitrión" and "Dominio". A button labeled "Añadir" is positioned below these fields. At the bottom of the main content area, there is a section titled "Tareas a realizar" with a list of actions: "reconectar", "guardar", and "desconectar". At the very bottom of the interface, there are two buttons: "Ejecutar" and "Desconectar", with an empty text input field between them.

Seleccionamos la opción de generar una nueva llave y proporcionamos el anfitrión y el Dominio.

Privilegios Mensajes SNMP SSH NTP Vistas ACL Auto asegurar

Información General Contraseña Líneas Usuarios Conexión Acceso Niveles

### Llave RSA

**Mostrar llave actual**

No hay llave actualmente

Generar nueva llave.

Tamaño de módulo

**Anfitrión**

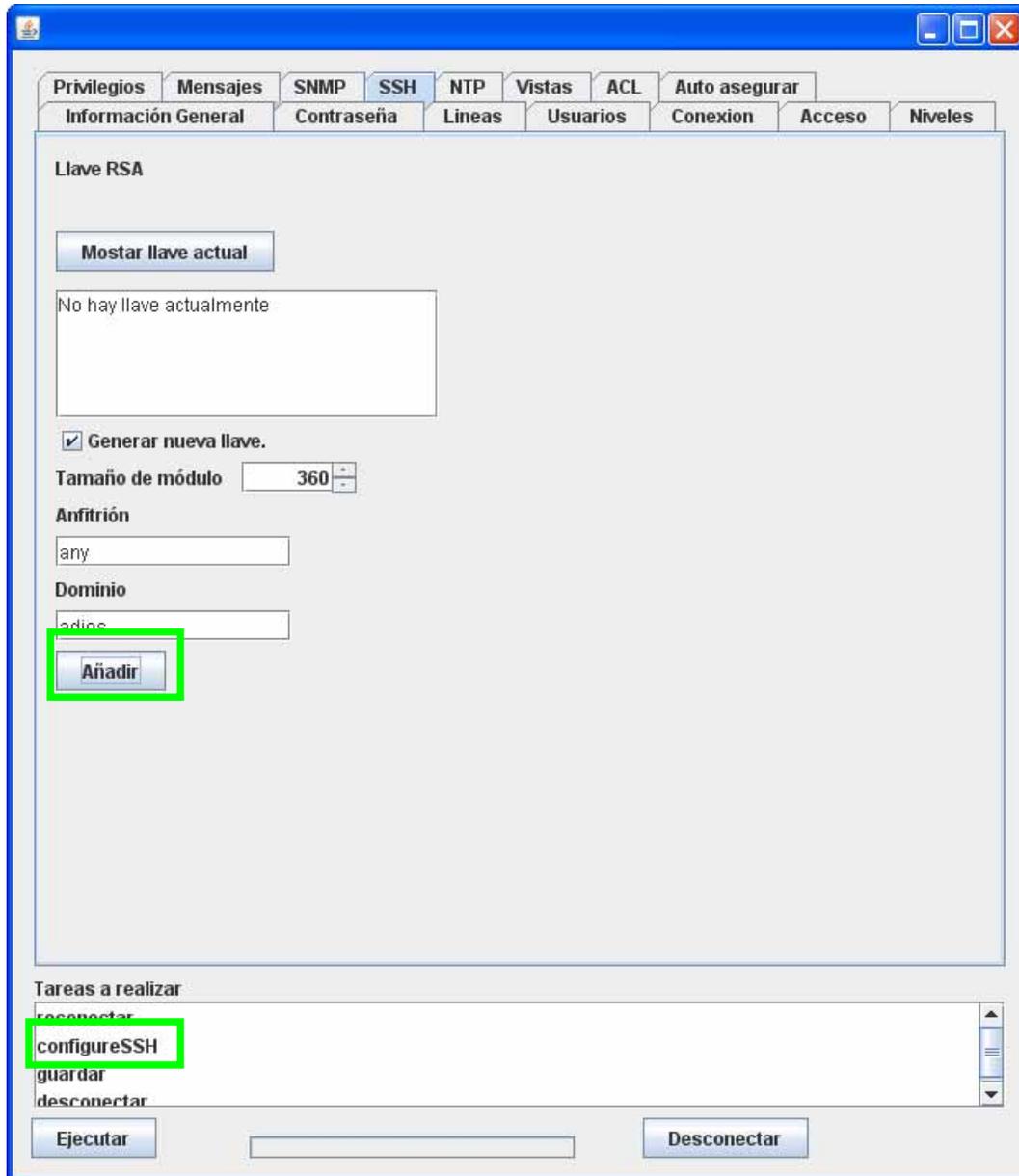
**Dominio**

**Añadir**

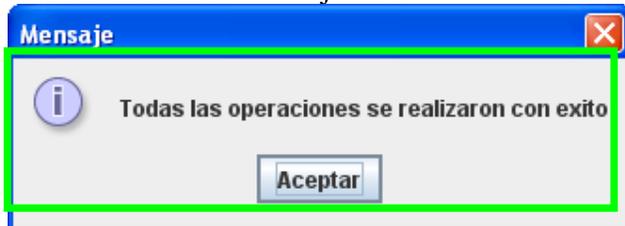
**Tareas a realizar**  
reconectar  
guardar  
desconectar

**Ejecutar**  **Desconectar**

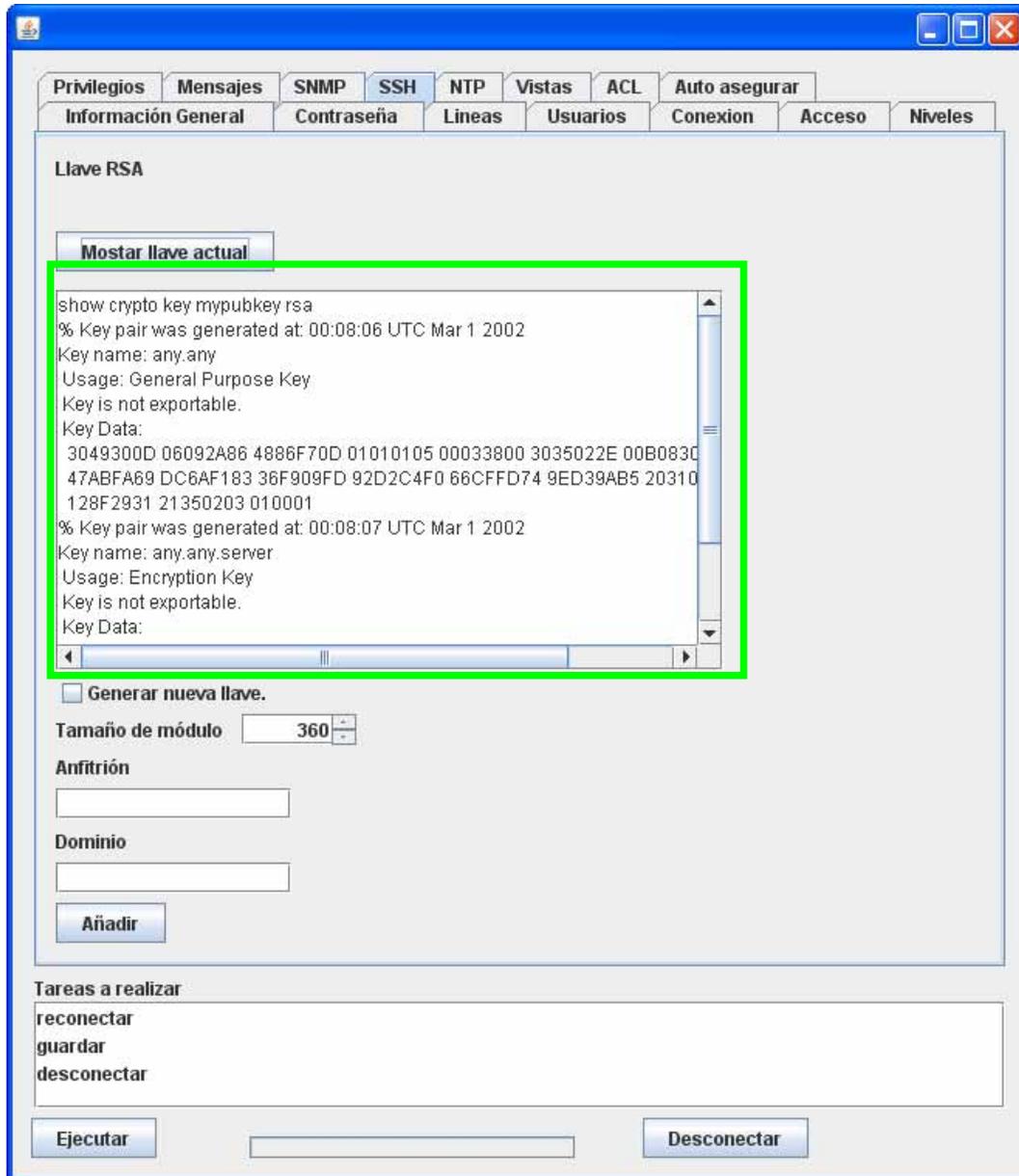
Presionamos el botón de añadir.



Presionamos el botón de ejecutar.



Presionamos el botón demostrar debe actuar para ver la llave que acabamos de generar.



Observamos la misma llave dentro de la configuración del Router (ruteador).

```
Dynamips(0): R1 , Console port
exec-timeout 2 2
line aux 0
exec-timeout 4 4
line vty 0
exec-timeout 1 1
password linea
line vty 1
password linea
line vty 2
exec-timeout 3 2
password jijiji
line vty 3
password hola
line vty 4
password adios
parser view vista1
secret 5 $1$TVOB$E5HmQ3ZnI4zGeF1S2ch2C.
commands exec include all ping
commands exec include show running-config
commands exec include show
commands exec exclude debug
!
!
ntp server 10.1.1.3 prefer
ntp server 10.1.1.2 prefer
end

any#show cry
any#show crypto k
any#show crypto key my
any#show crypto key mypubkey rs
any#show crypto key mypubkey rsa
% Key pair was generated at: 00:08:06 UTC Mar 1 2002
Key name: any.any
Usage: General Purpose Key
Key is not exportable.
Key Data:
 3049300D 06092A86 4886F70D 01010105 00033800 3035022E 00B08307 7EF22DAB
 47ABFA69 DC6AF183 36F909FD 92D2C4F0 66CFFD74 9ED39AB5 20310A7F C6F025E8
 128F2931 21350203 010001
% Key pair was generated at: 00:08:07 UTC Mar 1 2002
Key name: any.any.server
Usage: Encryption Key
Key is not exportable.
Key Data:
 307C300D 06092A86 4886F70D 01010105 00036B00 30680261 00CBBA35 DC62D414
 44A43362 7053E47D D22EE8EE 08A371E4 F2CA7EFD 8B8956EC D5650280 30B01B4F
 A4929119 9B49BAB7 87657861 E489BB83 B2F8C44C AFBAB8412 32F50898 CDB250F1
 EBC0F5F7 FE9957AA 27F189E3 87278A29 3FBF07B0 D16ADE12 A5020301 0001
any#
```

## Prueba 023: creación de comunidad SNMP

Punto de partida.

Nombre de la comunidad  tipo de comunidad RO

Comunidades

Vistas disponibles en router.

vista

Listas disponibles en router

10

| Comunidad | Vista | Tipo | ACL |
|-----------|-------|------|-----|
| escritura | vista | RW   |     |
| lectura   | vista | RO   |     |

Añadir Eliminar

Tareas a realizar

reconectar  
guardar  
desconectar

Ejecutar  Desconectar

Observamos de la configuración dentro del Router (ruteador).

```
Dynamips(0): R1 , Console port
!
interface Ethernet0
 no ip address
 shutdown
 half-duplex
!
interface FastEthernet0
 ip address 10.1.1.1 255.255.255.0
 speed auto
!
interface Serial0
 no ip address
 shutdown
!
interface Serial1
 no ip address
 shutdown
!
ip forward-protocol nd
!
no ip http server
no ip http secure-server
!
access-list 10 permit 10.1.1.2
access-list 10 permit 10.1.1.3
access-list 10 permit 10.1.1.4
access-list 10 deny 10.1.1.5
access-list 10 deny any log
snmp-server view vista iso.* included
snmp-server view vista system included
snmp-server view vista cisco included
snmp-server community lectura view vista RO
snmp-server community escritura view vista RW
snmp-server enable traps tty
!
control-plane
!
privilege exec all level 5 ping
privilege exec level 2 auto
privilege exec level 4 debug
!
line con 0
 exec-timeout 2 2
line aux 0
 exec-timeout 4 4
line vty 0
--More--
```

Proporcionamos el hombre de la comunidad.

Privilegios Mensajes **SNMP** SSH NTP Vistas ACL Auto asegurar

Información General Contraseña Lineas Usuarios Conexion Acceso Niveles

Nombre de la comunidad:  tipo de comunidad: RO

Comunidades

Vistas disponibles en router.

vista

Listas disponibles en router

10

| Comunidad | Vista | Tipo | ACL |
|-----------|-------|------|-----|
| escritura | vista | RW   |     |
| lectura   | vista | RO   |     |

Añadir Eliminar

Tareas a realizar

reconectar  
guardar  
desconectar

Ejecutar  Desconectar

Proporcionamos el tipo de la comunidad.

Privilegios Mensajes **SNMP** SSH NTP Vistas ACL Auto asegurar

Información General Contraseña Líneas Usuarios Conexión Acceso Niveles

Nombre de la comunidad: com tipo de comunidad: **RW**

Comunidades

Vistas disponibles en router.

vista

Listas disponibles en router

10

| Comunidad | Vista | Tipo | ACL |
|-----------|-------|------|-----|
| escritura | vista | RW   |     |
| lectura   | vista | RO   |     |

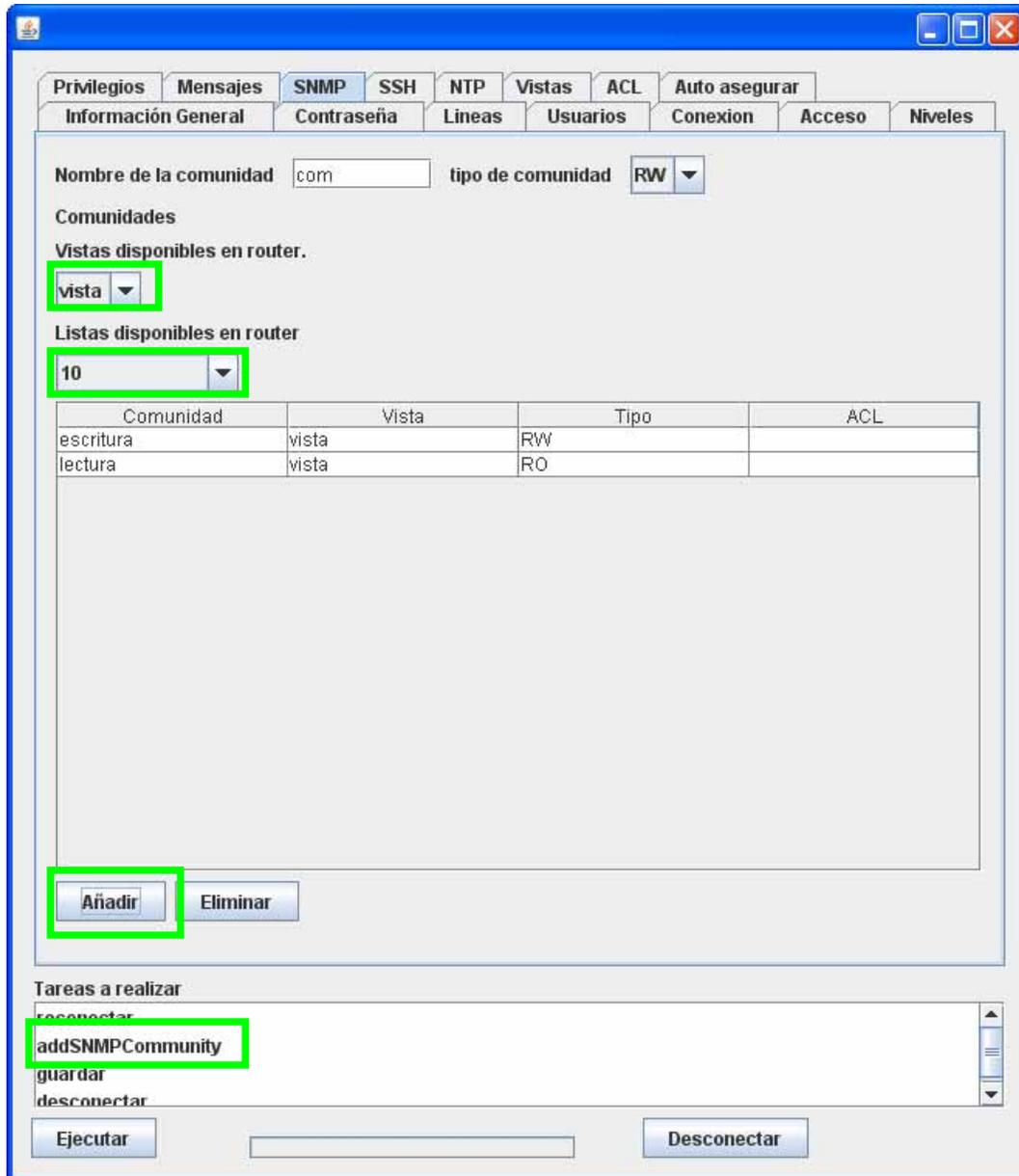
Añadir Eliminar

Tareas a realizar

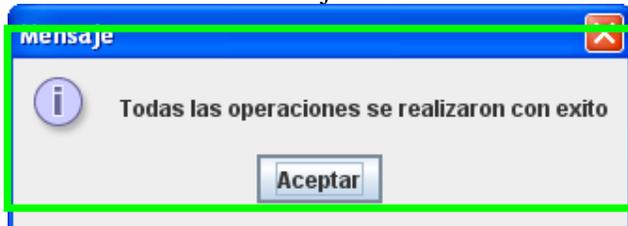
reconectar  
guardar  
desconectar

Ejecutar Desconectar

Seleccionamos la vista y la lista disponibles dentro del Router (ruteador).  
Presionamos el botón de añadir.



Presionamos el botón de ejecutar.



Observamos los cambios en la interfaz.

Privilegios Mensajes **SNMP** SSH NTP Vistas ACL Auto asegurar

Información General Contraseña Líneas Usuarios Conexión Acceso Niveles

Nombre de la comunidad  tipo de comunidad RW

Comunidades

Vistas disponibles en router.

Listas disponibles en router

| Comunidad | Vista | Tipo | ACL |
|-----------|-------|------|-----|
| com       | vista | RW   | 10  |
| escritura | vista | RW   |     |
| lectura   | vista | RO   |     |

Añadir Eliminar

Tareas a realizar

desconectarOK

reconectar

guardar

desconectar

Ejecutar  Desconectar

Comprobamos la información dentro de la configuración del Router (ruteador).

```
Dynamips(0): R1, Console port
!
interface Ethernet0
 no ip address
 shutdown
 half-duplex
!
interface FastEthernet0
 ip address 10.1.1.1 255.255.255.0
 speed auto
!
interface Serial0
 no ip address
 shutdown
!
interface Serial1
 no ip address
 shutdown
!
ip forward-protocol nd
!
no ip http server
no ip http secure-server
!
access-list 10 permit 10.1.1.2
access-list 10 permit 10.1.1.3
access-list 10 permit 10.1.1.4
access-list 10 deny 10.1.1.5
access-list 10 deny any log
snmp-server view vista iso.* included
snmp-server view vista system included
snmp-server view vista cisco included
snmp-server community lectura view vista RO
snmp-server community escritura view vista RW
snmp-server community com view vista RW 10
snmp-server enable traps tty
!
control-plane
!
privilege exec all level 5 ping
privilege exec level 2 auto
privilege exec level 4 debug
!
line con 0
 exec-timeout 2 2
line aux 0
 exec-timeout 4 4
--More--
```

Prueba 025: Eliminar comunidad SNMP  
Punto de partida.

Privilegios Mensajes **SNMP** SSH NTP Vistas ACL Auto asegurar

Información General Contraseña Líneas Usuarios Conexión Acceso Niveles

Nombre de la comunidad  tipo de comunidad **RW** ▼

Comunidades

Vistas disponibles en router.

**vista** ▼

Listas disponibles en router

**10** ▼

| Comunidad | Vista | Tipo | ACL |
|-----------|-------|------|-----|
| com       | vista | RW   | 10  |
| escritura | vista | RW   |     |
| lectura   | vista | RO   |     |

**Añadir** **Eliminar**

Tareas a realizar

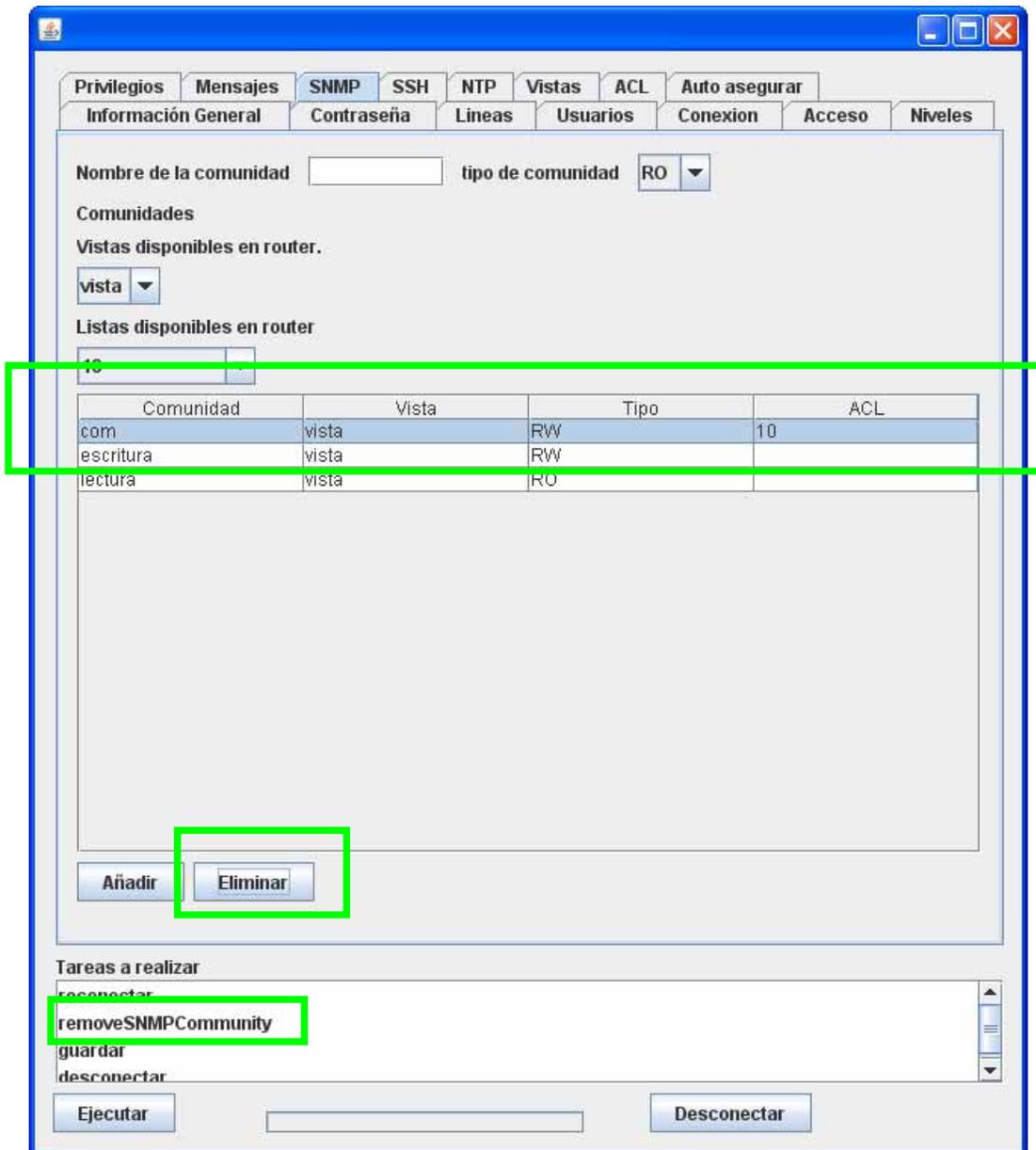
desconectarOK  
reconectar  
guardar  
desconectar

**Ejecutar**  **Desconectar**

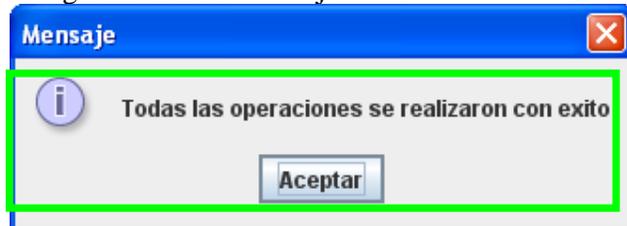
Comprobamos la misma información dentro de la configuración del Router (ruteador).

```
Dynamips(0): R1 , Console port
access-list 10 permit 10.1.1.2
access-list 10 permit 10.1.1.3
access-list 10 permit 10.1.1.4
access-list 10 deny 10.1.1.5
access-list 10 deny any log
snmp-server view vista iso.* included
snmp-server view vista system included
snmp-server view vista cisco included
snmp-server community lectura view vista RO
snmp-server community escritura view vista RW
snmp-server community com view vista RW 10
snmp-server enable traps tty
!
control-plane
!
privilege exec all level 5 ping
privilege exec level 2 auto
privilege exec level 4 debug
!
line con 0
  exec-timeout 2 2
line aux 0
  exec-timeout 4 4
line vty 0
  exec-timeout 1 1
  password linea
line vty 1
  password linea
line vty 2
  exec-timeout 3 2
  password jijiji
line vty 3
  password hola
line vty 4
  password adios
parser view vista1
secret 5 $1$TVOB$E5HmQ3ZnI4zGeF1S2ch2C.
commands exec include all ping
commands exec include show running-config
commands exec include show
commands exec exclude debug
!
!
ntp server 10.1.1.3 prefer
ntp server 10.1.1.2 prefer
end
Router#
```

Seleccionamos una comunidad de la tabla de comunidades y presionamos el botón de eliminar.



Resignamos el botón de ejecutar.



Observamos los cambios dentro de la interfaz.

Privilegios Mensajes **SNMP** SSH NTP Vistas ACL Auto asegurar

Información General Contraseña Líneas Usuarios Conexión Acceso Niveles

Nombre de la comunidad  tipo de comunidad RO ▼

Comunidades

Vistas disponibles en router.

vista ▼

Listas disponibles en router

10

| Comunidad | Vista | Tipo | ACL |
|-----------|-------|------|-----|
| escritura | vista | RW   |     |
| lectura   | vista | RO   |     |

Añadir Eliminar

Tareas a realizar

desconectarOK

reconectar

guardar

desconectar

Ejecutar  Desconectar

Confirmamos los datos dentro de la configuración del Router (ruteador).

```
Dynamips(0): R1 , Console port
!
interface Ethernet0
 no ip address
 shutdown
 half-duplex
!
interface FastEthernet0
 ip address 10.1.1.1 255.255.255.0
 speed auto
!
interface Serial0
 no ip address
 shutdown
!
interface Serial1
 no ip address
 shutdown
!
ip forward-protocol nd
!
no ip http server
no ip http secure-server
!
access-list 10 permit 10.1.1.2
access-list 10 permit 10.1.1.3
access-list 10 permit 10.1.1.4
access-list 10 deny 10.1.1.5
access-list 10 deny any log
snmp-server view vista iso.* included
snmp-server view vista system included
snmp-server view vista cisco included
snmp-server community lectura view vista RO
snmp-server community escritura view vista RW
snmp-server enable traps tty
!
control-plane
!
privilege exec all level 5 ping
privilege exec level 2 auto
privilege exec level 4 debug
!
line con 0
 exec-timeout 2 2
line aux 0
 exec-timeout 4 4
line vty 0
--More--
```