



Instituto Politecnico Nacional



ESCOM “ESCUELA SUPERIOR DE CÓMPUTO”

ADMINISTRACIÓN DE SERVICIOS EN RED

EXAMEN 2 PARCIAL

PROFE: RICARDO MARTÍNEZ ROSALES

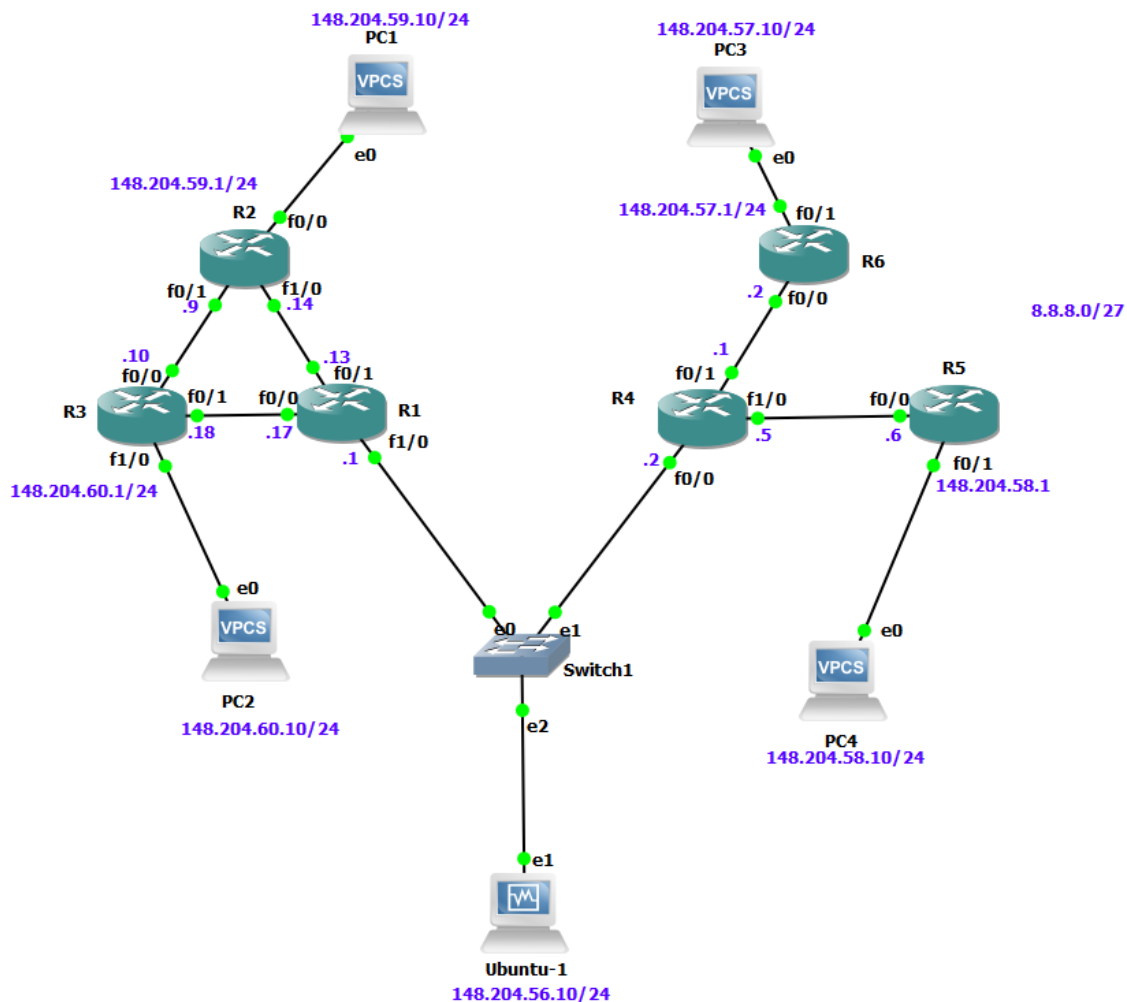
ALUMNO: Rojas Alvarado Luis Enrique

GRUPO: 4CM1

Objetivo

Realizar un programa en Python que funcione sobre el framework flask o equivalente para realizar operaciones de administración en redes básicas.

Implementar una base de datos de SQLAlchemy o equivalente sobre el framework elegido para almacenar información sobre la red de computadoras.



Configuración inicial

Todos los dispositivos finales son VPCS, menos la máquina virtual del centro (MV) que deberá de tener instalado Python y las APIs necesarias para la ejecución del programa y deberán de tener configurado su IP, máscara de subred y compuerta por defecto.

La subred para los 2 enlaces entre routers de cada sección corresponden a la 8.8.8.0/27 y debe de ser subneteada para su correcto funcionamiento.

Los enrutadores solo tendrán configurado las IP de sus interfaces con sus máscaras de subred respectivas, hostname único y una cuenta admin con clave de acceso admin con acceso de SSH y todos los permisos para acceso remoto. Tendrá el protocolo de enrutamiento dinámico OSPF.

```
R1#sh ip proto
Routing Protocol is "ospf 1"
  Outgoing update filter list for all interfaces is not set
  Incoming update filter list for all interfaces is not set
  Router ID 192.168.1.0
  It is an area border router
  Number of areas in this router is 2. 2 normal 0 stub 0 nssa
  Maximum path: 4
  Routing for Networks:
    8.8.8.12 0.0.0.3 area 1
    8.8.8.16 0.0.0.3 area 1
    148.204.56.0 0.0.0.255 area 0
  Reference bandwidth unit is 100 mbps
  Routing Information Sources:
    Gateway         Distance      Last Update
    192.168.2.0      110          00:41:00
    192.168.3.0      110          00:41:00
    192.168.4.0      110          00:41:00
  Distance: (default is 110)
```

R1#

```
R2#sh ip proto
Routing Protocol is "ospf 1"
  Outgoing update filter list for all interfaces is not set
  Incoming update filter list for all interfaces is not set
  Router ID 192.168.2.0
  Number of areas in this router is 1. 1 normal 0 stub 0 nssa
  Maximum path: 4
  Routing for Networks:
    8.8.8.8 0.0.0.3 area 1
    8.8.8.12 0.0.0.3 area 1
    148.204.59.0 0.0.0.255 area 1
  Reference bandwidth unit is 100 mbps
  Routing Information Sources:
    Gateway         Distance      Last Update
    192.168.1.0      110          00:42:19
    192.168.3.0      110          00:42:20
  Distance: (default is 110)
```

R2#

```
R3#sh ip proto
Routing Protocol is "ospf 1"
  Outgoing update filter list for all interfaces is not set
  Incoming update filter list for all interfaces is not set
  Router ID 192.168.3.0
  Number of areas in this router is 1. 1 normal 0 stub 0 nssa
  Maximum path: 4
  Routing for Networks:
    8.8.8.8 0.0.0.3 area 1
    8.8.8.16 0.0.0.3 area 1
    148.204.60.0 0.0.0.255 area 1
  Reference bandwidth unit is 100 mbps
  Routing Information Sources:
    Gateway         Distance      Last Update
    192.168.1.0      110          00:43:31
    192.168.2.0      110          00:43:31
  Distance: (default is 110)
```

R3#

```
R4#sh ip proto
Routing Protocol is "ospf 1"
  Outgoing update filter list for all interfaces is not set
  Incoming update filter list for all interfaces is not set
  Router ID 192.168.4.0
  It is an area border router
  Number of areas in this router is 2. 2 normal 0 stub 0 nssa
  Maximum path: 4
  Routing for Networks:
    8.8.8.0 0.0.0.3 area 2
    8.8.8.4 0.0.0.3 area 2
    148.204.56.0 0.0.0.255 area 0
  Reference bandwidth unit is 100 mbps
  Routing Information Sources:
    Gateway         Distance      Last Update
    192.168.1.0      110          00:43:55
    192.168.5.0      110          00:44:05
    192.168.6.0      110          00:44:05
  Distance: (default is 110)
```

R4#

```
R5#sh ip proto
Routing Protocol is "ospf 1"
  Outgoing update filter list for all interfaces is not set
  Incoming update filter list for all interfaces is not set
  Router ID 192.168.5.0
  Number of areas in this router is 1. 1 normal 0 stub 0 nssa
  Maximum path: 4
  Routing for Networks:
    8.8.8.4 0.0.0.3 area 2
    148.204.58.0 0.0.0.255 area 2
  Reference bandwidth unit is 100 mbps
  Routing Information Sources:
    Gateway         Distance      Last Update
    192.168.4.0      110          00:44:32
    192.168.6.0      110          00:44:32
  Distance: (default is 110)
```

R5#

```
R6#sh ip proto
Routing Protocol is "ospf 1"
  Outgoing update filter list for all interfaces is not set
  Incoming update filter list for all interfaces is not set
  Router ID 192.168.6.0
  Number of areas in this router is 1. 1 normal 0 stub 0 nssa
  Maximum path: 4
  Routing for Networks:
    8.8.8.0 0.0.0.3 area 2
    148.204.57.0 0.0.0.255 area 2
  Reference bandwidth unit is 100 mbps
  Routing Information Sources:
    Gateway         Distance      Last Update
    192.168.4.0      110          00:45:10
    192.168.5.0      110          00:45:10
  Distance: (default is 110)
```

R6#

Haciendo ping entre máquinas de GNS3 (PC1 y PC3)

```
PC1> ping 148.204.57.10
148.204.57.10 icmp_seq=1 timeout
84 bytes from 148.204.57.10 icmp_seq=2 ttl=60 time=79.888 ms
84 bytes from 148.204.57.10 icmp_seq=3 ttl=60 time=54.109 ms
84 bytes from 148.204.57.10 icmp_seq=4 ttl=60 time=54.378 ms
84 bytes from 148.204.57.10 icmp_seq=5 ttl=60 time=65.190 ms
PC1> █
```

Haciendo ping entre la máquina virtual a una máquina de GNS3.

```
luis@luis-VirtualBox:~$ ping 148.204.57.10
PING 148.204.57.10 (148.204.57.10) 56(84) bytes of data.
64 bytes from 148.204.57.10: icmp_seq=1 ttl=62 time=33.5 ms
64 bytes from 148.204.57.10: icmp_seq=2 ttl=62 time=39.5 ms
64 bytes from 148.204.57.10: icmp_seq=3 ttl=62 time=40.7 ms
64 bytes from 148.204.57.10: icmp_seq=4 ttl=62 time=41.5 ms
64 bytes from 148.204.57.10: icmp_seq=5 ttl=62 time=33.9 ms
64 bytes from 148.204.57.10: icmp_seq=6 ttl=62 time=36.7 ms
64 bytes from 148.204.57.10: icmp_seq=7 ttl=62 time=34.5 ms
^C
--- 148.204.57.10 ping statistics ---
7 packets transmitted, 7 received, 0% packet loss, time 6012ms
rtt min/avg/max/mdev = 33.534/37.200/41.533/3.127 ms
luis@luis-VirtualBox:~$
```

Primero que nada, debemos abrir el entorno virtual y ejecutar nuestro programa.

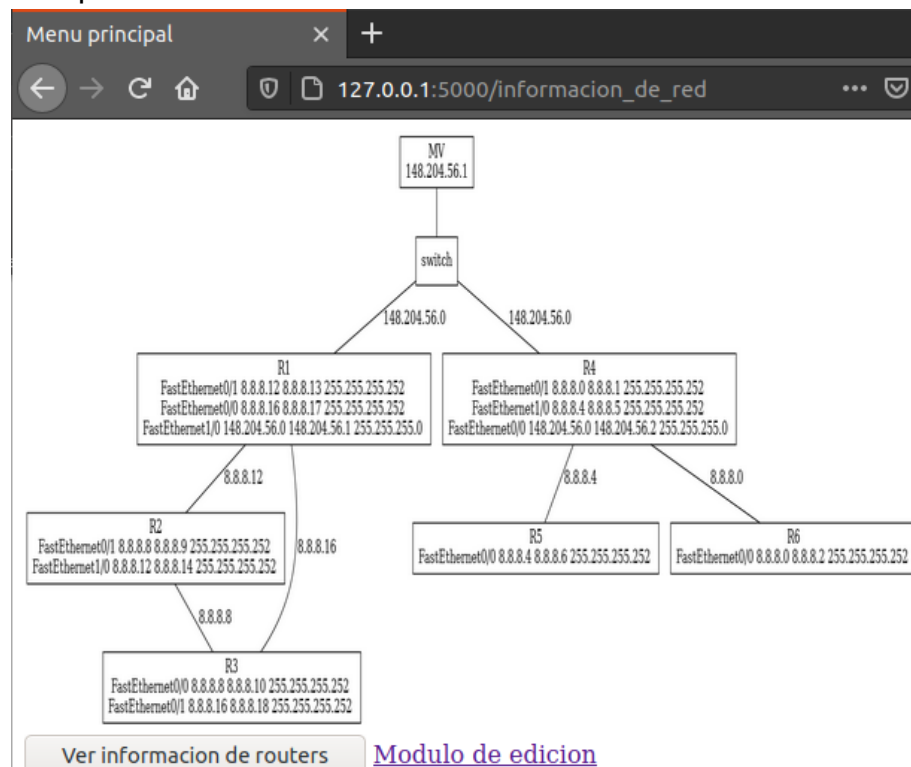
```
luis@luis-VirtualBox:~/Documentos$ ls
'Enrutamiento multiple'  Ex02  kk  Pipfile  Pipfile.lock  venv
luis@luis-VirtualBox:~/Documentos$ source venv/bin/activate
(venv) luis@luis-VirtualBox:~/Documentos$ cd Ex02/
(venv) luis@luis-VirtualBox:~/Documentos/Ex02$ python3
examen2.py static/ templates/
(venv) luis@luis-VirtualBox:~/Documentos/Ex02$ python3
examen2.py static/ templates/
(venv) luis@luis-VirtualBox:~/Documentos/Ex02$ python3 examen2.py
* Running on http://0.0.0.0:5000/
* Restarting with reloader
```

Ahora debemos de ingresar a una pestaña de cualquier navegador web por el puerto que nos marca flask. Cabe destacar que para la primera vez que se ejecuta el servidor en el directorio raíz "/", empieza a reconocer la puerta de enlace por defecto y empieza a analizar la raíz. En la parte superior podemos observar la dirección en la que se ejecuta y en la consola podemos observar que analiza la red entrando por las puertas de enlace (programa hecho en el primer parcial).

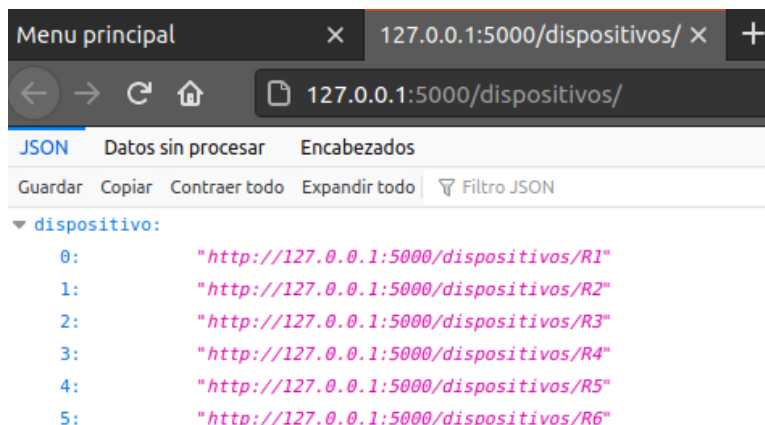
Agregando los comandos pertinentes para obtener la marca y el modelo del router, así como los usuarios y contraseñas para entrar con ssh.

```
Restaurar sesión x +
127.0.0.1:5000
luis@luis-VirtualBox: ~/Documentos/Ex02
examen2.py static/ templates/
(venv) luis@luis-VirtualBox:~/Documentos/Ex02$ python3 examen2.py
* Running on http://0.0.0.0:5000/
* Restarting with reloader
este es el default 148.204.56.1
conectando a 148.204.56.1
despues del try
este es prompt : R1
conectando a 148.204.56.2
despues del try
este es prompt : R4
conectando a 8.8.8.18
despues del try
este es prompt : R3
conectando a 8.8.8.14
despues del try
este es prompt : R2
conectando a 8.8.8.6
despues del try
este es prompt : R5
conectando a 8.8.8.2
despues del try
este es prompt : R6
127.0.0.1
```

Una vez terminada de analizar la topología tal como se hacía en el examen 1, se procede a dibujarla y presentarla en la página web. En la cual se redirecciona al directorio “/informacion_de_red “. En el cual funcionará como página principal en la que podemos ver la información de todos los dispositivos de red y podemos editar cualquier campo.

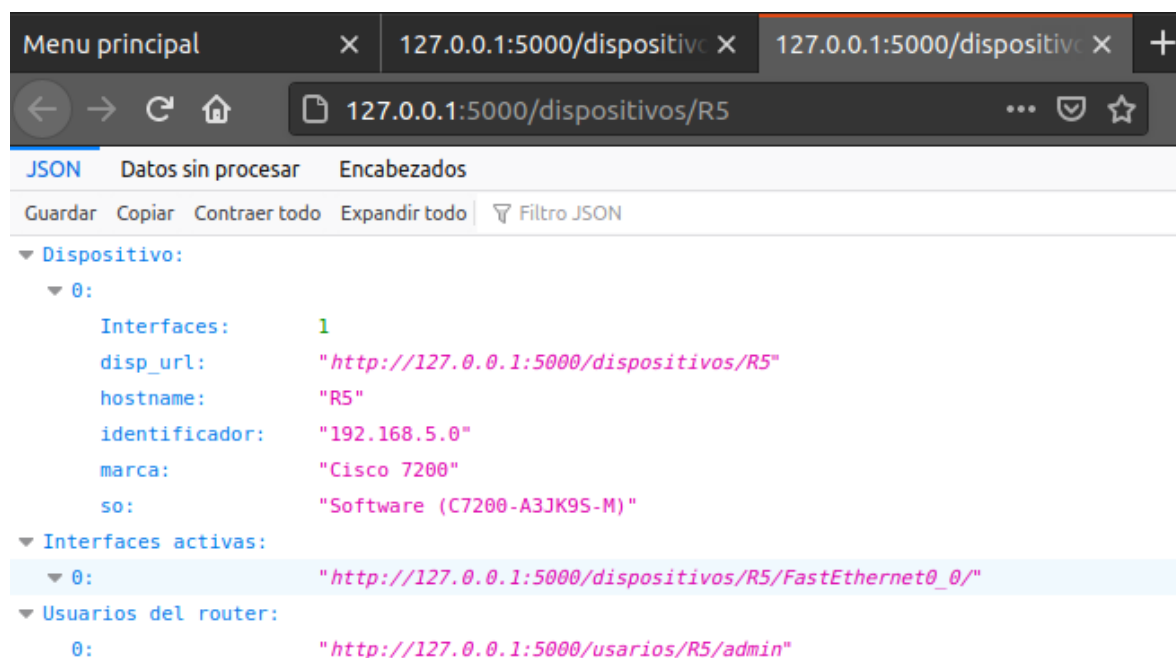


Primero con el módulo de ver información de routers. Si damos clic en ese botón, nos redireccionará al directorio “/dispositivos” en el cual nos desplegará un JSON con la información de todos los routers analizados anteriormente, en el cual podemos ver su información dando clic en cualquiera de estos links.



```
Menu principal x 127.0.0.1:5000/dispositivos/ x +
127.0.0.1:5000/dispositivos/
JSON Datos sin procesar Encabezados
Guardar Copiar Contraer todo Expandir todo Filtro JSON
▼ dispositivo:
0: "http://127.0.0.1:5000/dispositivos/R1"
1: "http://127.0.0.1:5000/dispositivos/R2"
2: "http://127.0.0.1:5000/dispositivos/R3"
3: "http://127.0.0.1:5000/dispositivos/R4"
4: "http://127.0.0.1:5000/dispositivos/R5"
5: "http://127.0.0.1:5000/dispositivos/R6"
```

Si damos clic en cualquiera de los links podemos ver la información del router. Nos redirecciona al directorio “dispositivos/<router>”. Y nos despliega la información del dispositivo, como las interfaces, el nombre de dispositivo, su identificador (o loopback), marca y sistema operativo, las interfaces activas y los usuarios en el router. Podemos ingresar a ver la información de las interfaces activas y los usuarios en el router y editarla, cabe destacar que mientras la edición, eliminación y nuevo usuario se ven reflejados tanto en la base de datos como en el dispositivo en GNS3, las edición de interfaces activas solo se ve reflejada en la base de datos.



```
Menu principal x 127.0.0.1:5000/dispositivos/R5 x 127.0.0.1:5000/dispositivos/ x +
127.0.0.1:5000/dispositivos/R5 ...
JSON Datos sin procesar Encabezados
Guardar Copiar Contraer todo Expandir todo Filtro JSON
▼ Dispositivo:
  ▼ 0:
    Interfaces: 1
    disp_url: "http://127.0.0.1:5000/dispositivos/R5"
    hostname: "R5"
    identificador: "192.168.5.0"
    marca: "Cisco 7200"
    so: "Software (C7200-A3JK9S-M)"
  ▼ Interfaces activas:
    ▼ 0: "http://127.0.0.1:5000/dispositivos/R5/FastEthernet0_0/"
  ▼ Usuarios del router:
    0: "http://127.0.0.1:5000/usuarios/R5/admin"
```

Consultando, editando y comprobando cambios en las interfaces activas. Los datos a modificar se llenan en los campos y cuando damos al botón editar se nos redirecciona de nuevo a la página de información del router con la información editada, de la misma manera, podemos volver a consultar lo que editamos.

hostname : R5
name : Fa0_0
IP de interfaz : 8.8.8.14
mascara : 255.255.255.222
subred : 8.8.8.13

Hostname:
Nombre :
Ip de Interfaz :
mascara:
subred :

127.0.0.1:5000/dispositivos/R5

JSON Datos sin procesar Encabezados

Guardar Copiar Contraer todo Expandir todo Filtro JSON

▼ Dispositivo:

▼ 0:

Interfaces: 1

disp_url: "http://127.0.0.1:5000/dispositivos/R5"

hostname: "R5"

identificador: "192.168.5.0"

marca: "Cisco 7200"

so: "Software (C7200-A3JK9S-M)"

▼ Interfaces activas:

0: "http://127.0.0.1:5000/dispositivos/R5/Fa0_0/"

▼ Usuarios del router:

0: "http://127.0.0.1:5000/usuarios/R5/admin"

Como podemos darnos cuenta, es presindible no usar diagonales en el nuevo nombre, ya que lo tomaría como si fuera otra dirección y nunca se encontraría la información que editamos.

En el caso de editar los usuarios del router, se ve reflejado en la base de datos como en el router de GNS3. Si damos clic en el link del usuario podemos ver la información del usuario que está actualmente activo en el router y podemos agregar otro, o eliminarlo.

127.0.0.1:5000/usuarios/R5/admin

hostname : R5
Usuario : admin
password : 104F0D140C19

Hostname:
Usuario :
Ip Nuevo password :

127.0.0.1:5000/dispositivos/R5

JSON Datos sin procesar Encabezados

Guardar Copiar Contraer todo Expandir todo Filtro JSON

▼ Dispositivo:

▼ 0:

Interfaces: 1

disp_url: "http://127.0.0.1:5000/dispositivos/R5"

hostname: "R5"

identificador: "192.168.5.0"

marca: "Cisco 7200"

so: "Software (C7200-A3JK9S-M)"

▼ Interfaces activas:

0: "http://127.0.0.1:5000/dispositivos/R5/Fa0_0/"

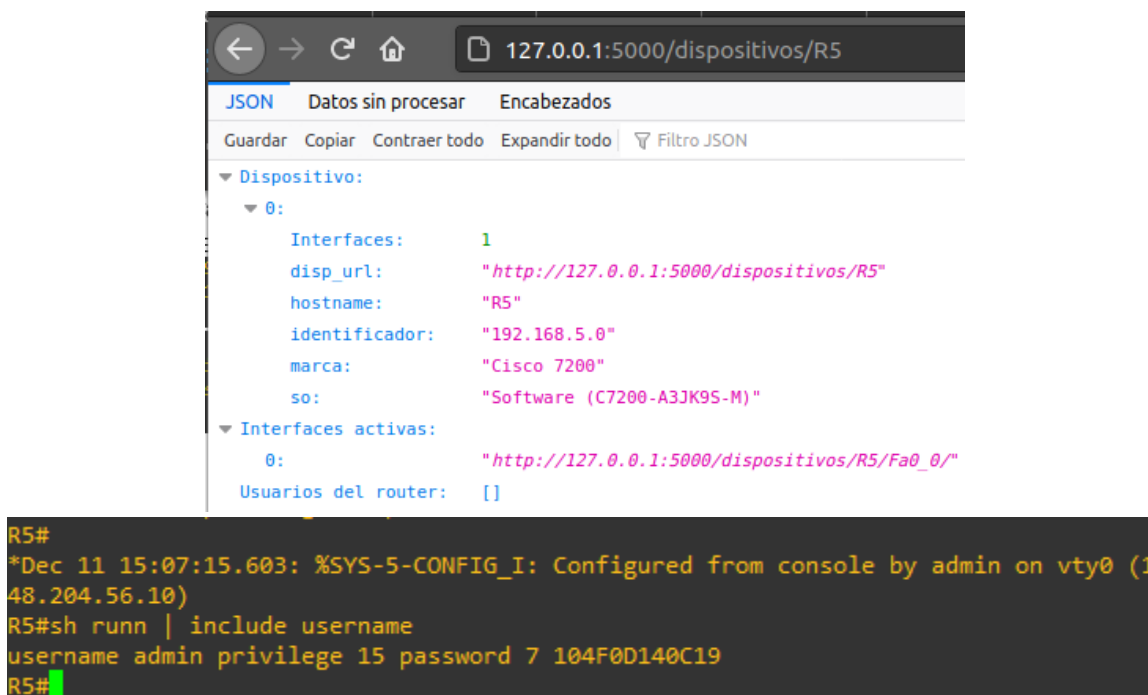
▼ Usuarios del router:

0: "http://127.0.0.1:5000/usuarios/R5/luis"

El usuario que agregamos tiene el nombre de luis y especificamos la direccion ip del router para acceder con el usuario admin. Si vamos a la consola del router podemos observar que se ha hecho una modificación remota.

```
*Dec 11 14:59:04.675: %SYS-5-CONFIG_I: Configured from console by admin on vty0
(148.204.56.10)
R5#
R5#sh runn | include username
username admin privilege 15 password 7 104F0D140C19
username luis privilege 15 password 7 055A52576F1E1E5D57504F5C5A
R5#
```

Entonces comprobamos con el comando `show running-config | include username` para revisar que la creación del nuevo usuario ha sido exitosa. Si queremos eliminarlo entonces damos clic en eliminar el usuario. Vemos que en la página de información del router ya no existen usuarios. Y lo comprobamos en el router GNS3.



The screenshot shows a web application interface with a JSON view of router data. The JSON structure is as follows:

```

{
  "Dispositivo": {
    "0": {
      "Interfaces": 1,
      "disp_url": "http://127.0.0.1:5000/dispositivos/R5",
      "hostname": "R5",
      "identificador": "192.168.5.0",
      "marca": "Cisco 7200",
      "so": "Software (C7200-A3JK9S-M)"
    }
  },
  "Interfaces activas": {
    "0": "http://127.0.0.1:5000/dispositivos/R5/Fa0_0/"
  },
  "Usuarios del router": []
}

```

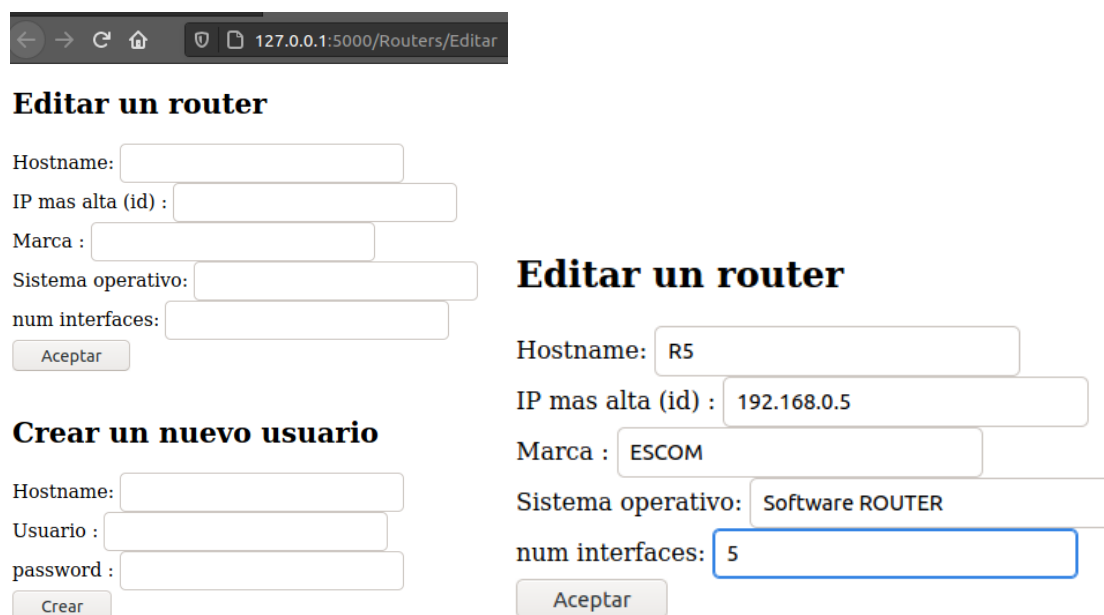
Below the JSON view, a terminal window shows the following commands and output:

```

R5#
*Dec 11 15:07:15.603: %SYS-5-CONFIG_I: Configured from console by admin on vty0 (148.204.56.10)
R5#sh runn | include username
username admin privilege 15 password 7 104F0D140C19
R5#

```

Regresando al menú principal, si damos clic en el módulo de edición, nos redirecciona al directorio `/Routers/Editar` en el que podemos editar la información de un router y verla representada en la base de datos (nos redireccionará a la página de información del router) y crear un usuario.

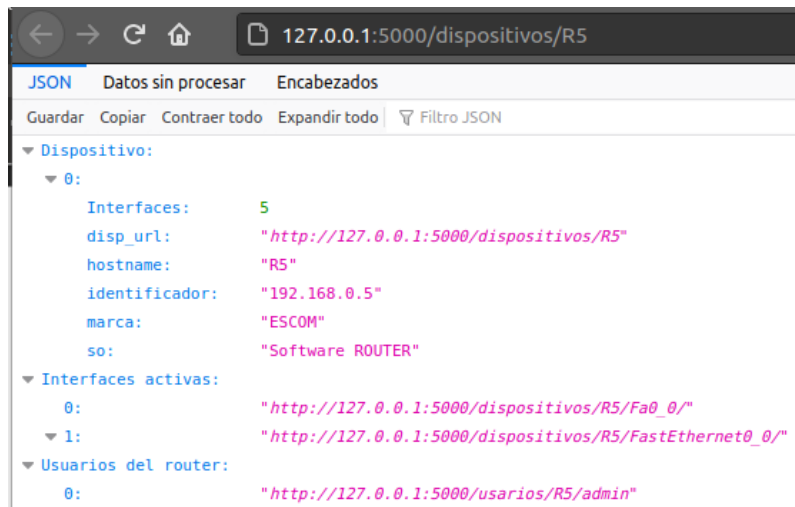


The screenshot shows the 'Editar un router' web form. The form has a title 'Editar un router' and a URL bar showing `127.0.0.1:5000/Routers/Editar`. The form contains the following fields and buttons:

- Hostname:
- IP mas alta (id) :
- Marca :
- Sistema operativo:
- num interfaces:
-

Below the form, there is a section titled 'Crear un nuevo usuario' with the following fields and button:

- Hostname:
- Usuario :
- password :
-

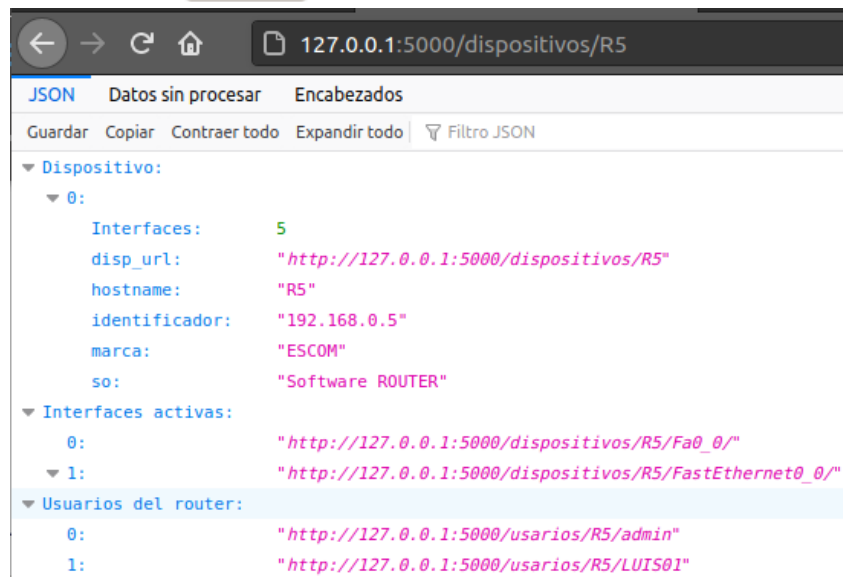


Crear un nuevo usuario

Hostname:

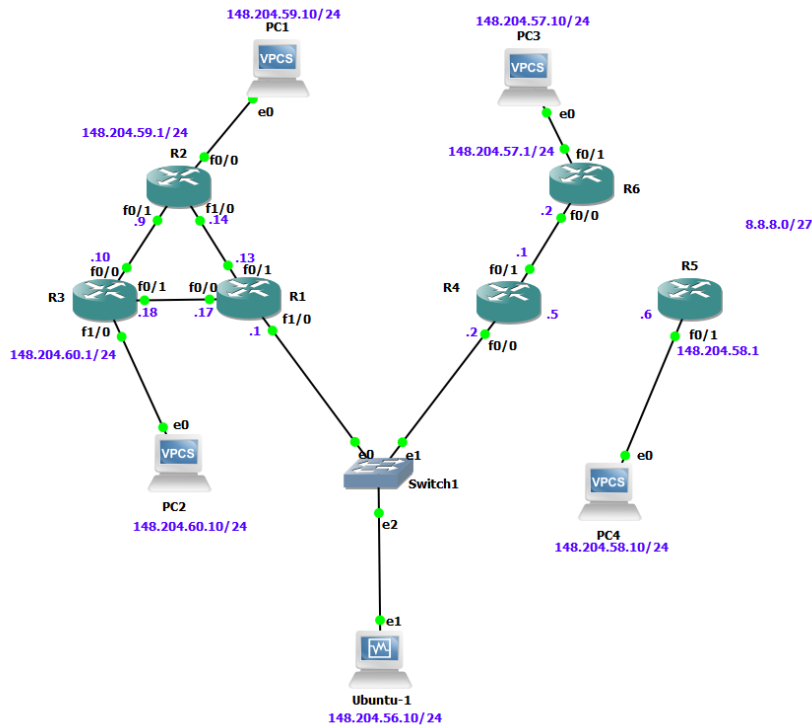
Usuario :

password :

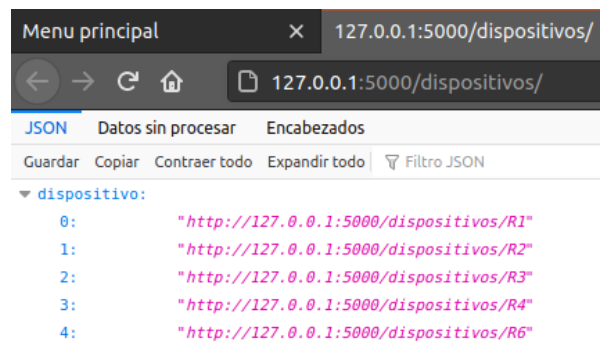
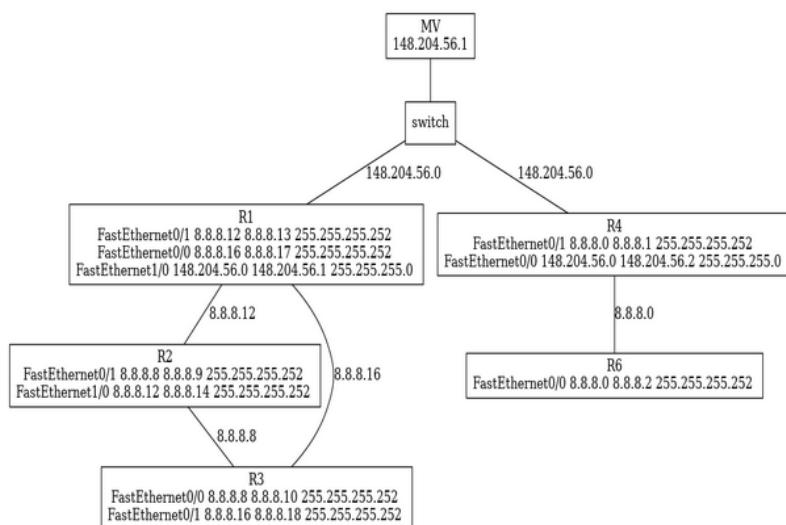
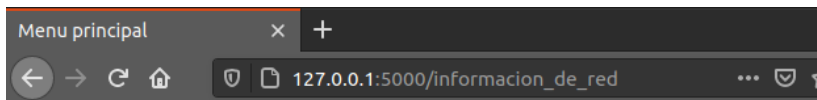


```
R5#
*Dec 11 15:23:24.667: %SYS-5-CONFIG_I: Configured from console by admin on vty0 (148.204.56.10)
R5#sh runn | include username
username admin privilege 15 password 7 104F0D140C19
username LUIS01 privilege 15 password 7 1235303E215B5D
R5#
```

Cabe destacar que los usuarios que agreguemos se reflejan en el router y es posible acceder al router mediante el usuario que acabamos de crear con ssh.



Ahora se hará la prueba con la topología modificada, eliminando una de las interfaces, por lo que en el análisis de la topología debería tener la modificación.



Se puede ver que tanto en la información de la topología como en la exploración de la misma, la interfaz que eliminamos no aparece en ambas.