

Laboratorio TDR2

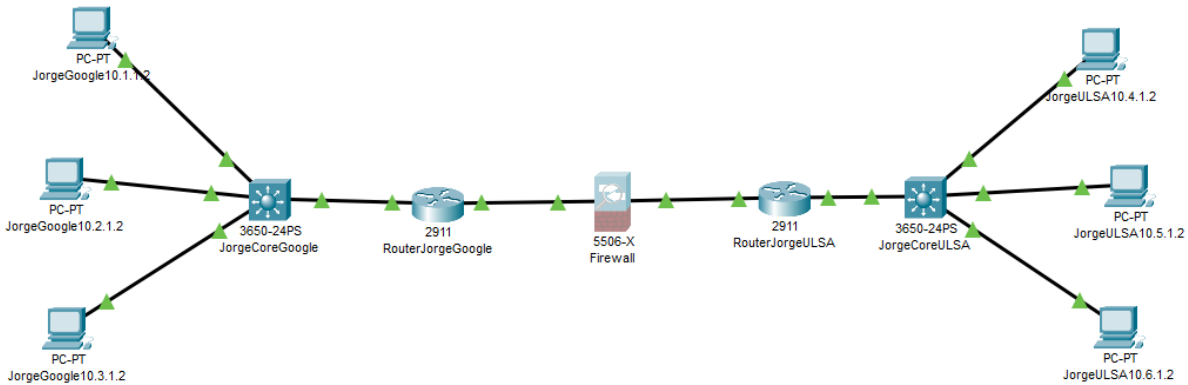
Parcial 1

Nombre: Jorge Parra Hidalgo

Carrera: ITIT

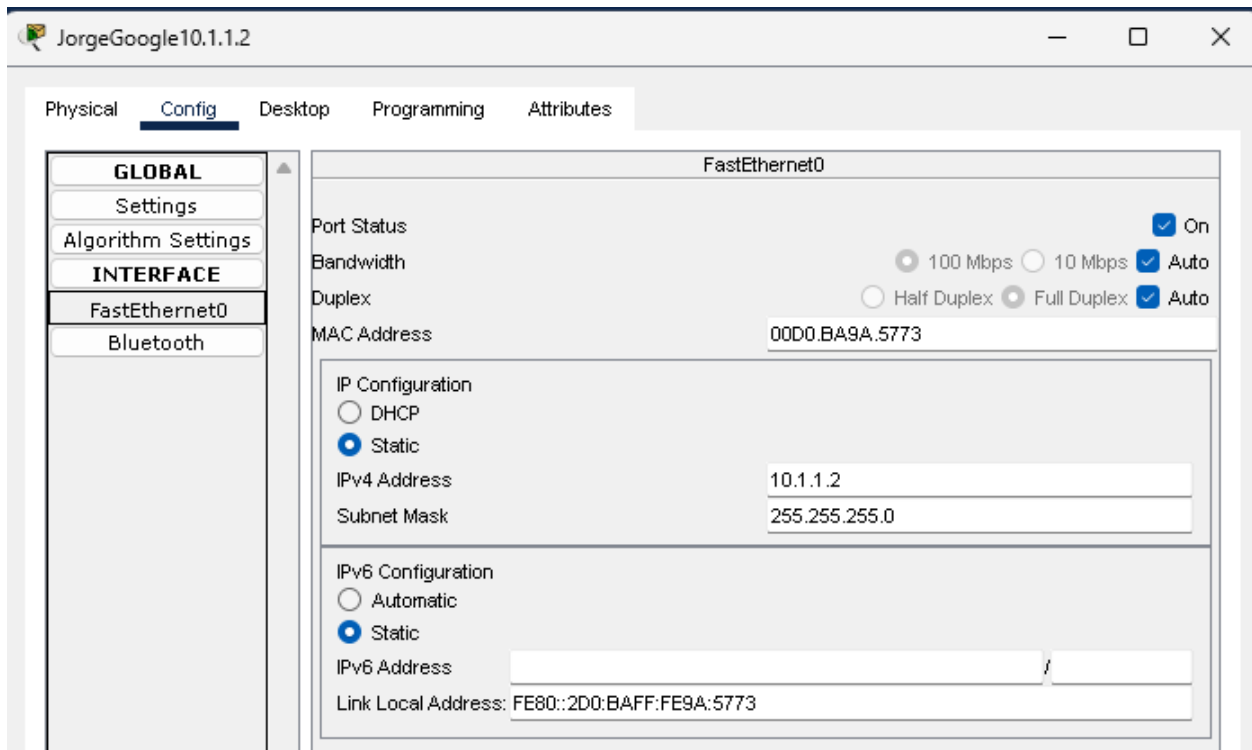
Matricula: 13104

Fecha: 9 de septiembre del 2024



Paso 1: Configurar IPs de PCs

Para comenzar vamos a establecer las IPs de las PCs y del lado de Google serán las IPs: 10.1.1.2, 10.2.1.2 y 10.3.1.2 y del lado de ULSA se continuará con 10.4.1.2, 10.5.1.2 y 10.6.1.2 y todos con una mascarará de 255.255.255.0



Paso 2: Configurar Core Switch

Primero es esencial poner un usuario y contraseña ya que cuando queramos abrir el telnet tengamos uno en mi caso escribimos: **username jorge password 123**

❓ username jorge: Define un nuevo usuario con el nombre de usuario "jorge".

❓ password 0 123: Establece la contraseña para el usuario "jorge". El "0" antes de la contraseña indica que la contraseña es de texto claro y no está cifrada. La contraseña en este caso es "123".

Luego procederemos a escribir los siguientes comandos:

interface GigabitEthernet1/0/1

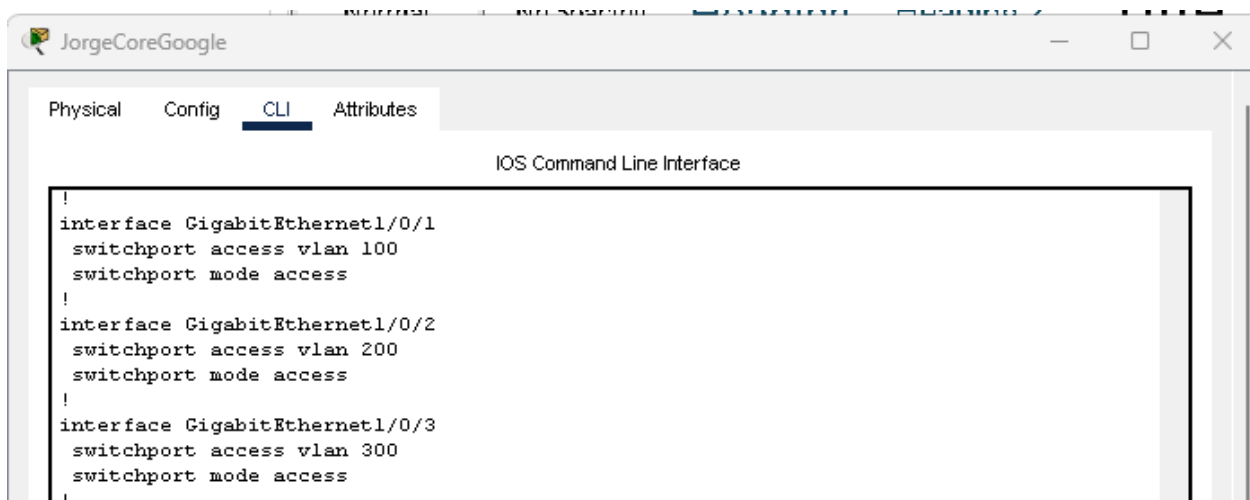
switchport access vlan 100

switchport mode Access

Estos comandos se usan para configurar una interfaz de switch en un dispositivo Cisco para que funcione en modo de acceso con una VLAN específica. Aquí está el desglose de cada comando:

- interface GigabitEthernet1/0/1: Este comando selecciona la interfaz GigabitEthernet1/0/1 en el switch para su configuración.
- switchport access vlan 100: Asigna la VLAN 100 a la interfaz seleccionada. Esto significa que el tráfico que pase por esta interfaz se etiquetará con la VLAN 100 y se tratará como parte de esa VLAN.
- switchport mode access: Configura la interfaz en modo de acceso. En el modo de acceso, la interfaz solo puede estar asociada con una sola VLAN a la vez, que en este caso es la VLAN 100. El modo de acceso es típico para interfaces que conectan dispositivos finales como computadoras y impresoras.

Luego así sucesivamente con el puerto 2 seria vlan 200 y así sucesivamente tanto de lado de Google como de ULSA.



```
!
interface GigabitEthernet1/0/1
  switchport access vlan 100
  switchport mode access
!
interface GigabitEthernet1/0/2
  switchport access vlan 200
  switchport mode access
!
interface GigabitEthernet1/0/3
  switchport access vlan 300
  switchport mode access
!
```

Paso 3: Configurar interfaces de VLAN

interface Vlan100

ip address 10.1.1.1 255.255.255.0

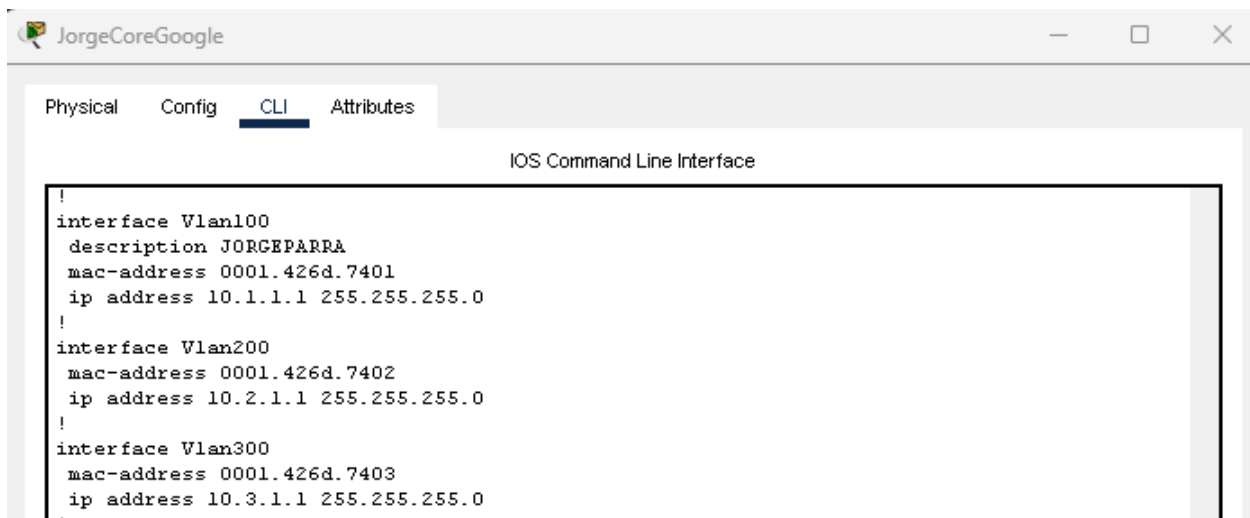
Al hacer esto estamos entrando a la interfaz del vlan que queremos y le ponemos la ip a la pc que es así como su mascara, esto se repite con cada vlan creada (6 en total)

interface Vlan100:

- Este comando se usa en la configuración de un switch de Cisco para entrar en el modo de configuración de una interfaz VLAN específica. En este caso, Vlan100 se refiere a la VLAN número 100. Al usar este comando, estás configurando una interfaz VLAN virtual en el switch.

ip address 10.1.1.1 255.255.255.0:

- Este comando se utiliza para asignar una dirección IP y una máscara de subred a la interfaz VLAN. En este caso, estás configurando la dirección IP 10.1.1.1 con la máscara de subred 255.255.255.0 para la interfaz VLAN 100.
-



```
!
interface Vlan100
  description JORGEPARRA
  mac-address 0001.426d.7401
  ip address 10.1.1.1 255.255.255.0
!
interface Vlan200
  mac-address 0001.426d.7402
  ip address 10.2.1.1 255.255.255.0
!
interface Vlan300
  mac-address 0001.426d.7403
  ip address 10.3.1.1 255.255.255.0
!
```

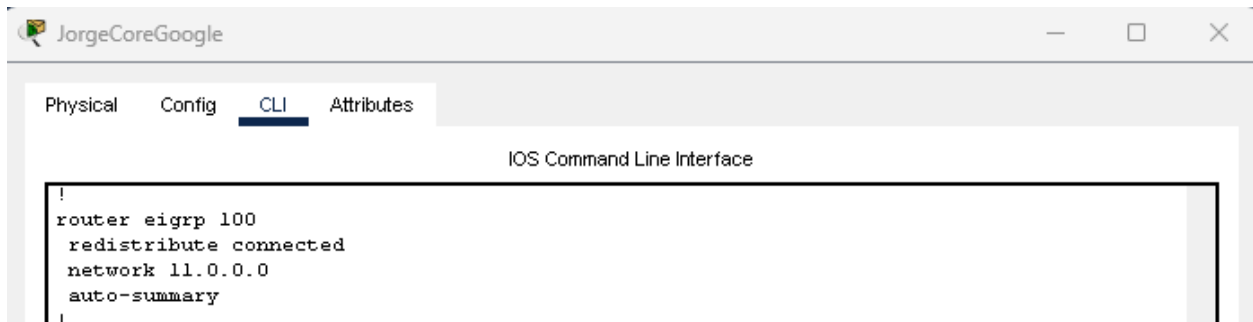
Paso 4: Conexión EIGRP

router eigrp 100

redistribute connected

network 11.0.0.0

En caso del primer switch es network 11.0.0.0 ya que este va a usar la ip 11.11.11.1 para conectarse al router y este tendrá la 11.11.11.2 para así conectarse por medio de una WAN por EIGRP. En caso del switch de ULSA a su router seria igual solo que con la ip 14.0.0.0



```
!
router eigrp 100
 redistribute connected
 network 11.0.0.0
 auto-summary
!
```

router eigrp 100:

- Entra en el modo de configuración de EIGRP para el proceso AS 100. Esto configura el router para utilizar EIGRP con ese número de AS.

redistribute connected:

- Anuncia las rutas de las interfaces conectadas directamente al proceso EIGRP. Es útil cuando se quieren incluir todas las redes conectadas directamente en el proceso de EIGRP.

network 11.0.0.0:

- Configura EIGRP para incluir todas las interfaces que pertenezcan a la red 11.0.0.0 en el proceso de enrutamiento. EIGRP automáticamente identifica las interfaces que coinciden con esta red y las incluye en el proceso de enrutamiento.

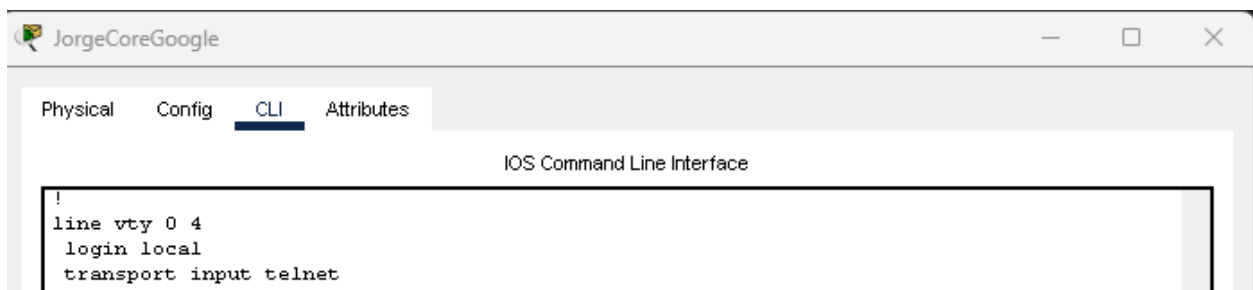
Paso 5: Telnet

line vty 0 4

login local

transport input telnet

Estos comandos son para poder acceder via telnet.



```
!
line vty 0 4
 login local
 transport input telnet
!
```

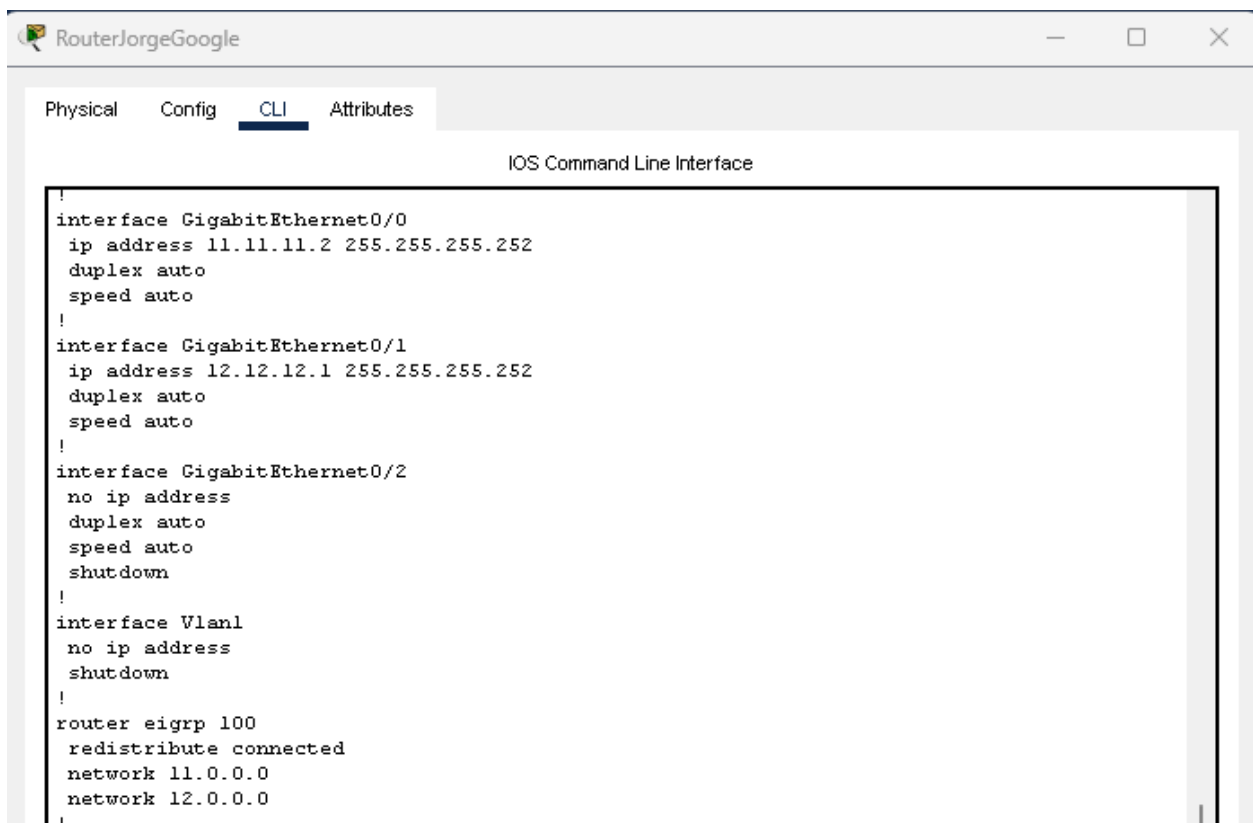
line vty 0 4: Configura las líneas VTY 0 a 4 para conexiones remotas.

login local: Utiliza la autenticación local basada en usuarios definidos en el dispositivo.

transport input telnet: Permite el acceso remoto mediante Telnet en las líneas VTY.

Paso 6: Configurar routers

Los routers es simplemente conectar las WANS de switch a router y de router a firewall tanto de ulsa como de Google. En mi caso elegí 11.0.0.0-14.0.0.0 para las WAN y el comando de EIGRP para poder conectarse.



```
!
interface GigabitEthernet0/0
 ip address 11.11.11.2 255.255.255.252
 duplex auto
 speed auto
!
interface GigabitEthernet0/1
 ip address 12.12.12.1 255.255.255.252
 duplex auto
 speed auto
!
interface GigabitEthernet0/2
 no ip address
 duplex auto
 speed auto
 shutdown
!
interface Vlan1
 no ip address
 shutdown
!
router eigrp 100
 redistribute connected
 network 11.0.0.0
 network 12.0.0.0
!
```

Los comandos ya están explicados previamente, simplemente cambia la dirección IP asignada.

Paso 7: Configurar firewall

Primero se pone la conexión WAN hacia los routers como ya previamente se menciona

```
interface GigabitEthernet1/1
 nameif GOOGLE
 security-level 10
 ip address 12.12.12.2 255.255.255.252
!
interface GigabitEthernet1/2
 nameif ULSA
 security-level 10
 ip address 13.13.13.2 255.255.255.252
!
```

Posteriormente se escribe los siguientes comandos

access-list GOOGLE extended permit tcp host 10.1.1.2 host 10.4.1.1 eq telnet

access-list GOOGLE extended permit tcp host 10.2.1.2 host 10.5.1.1 eq telnet

access-list GOOGLE extended permit tcp host 10.3.1.2 host 10.6.1.1 eq telnet

access-list GOOGLE extended deny tcp host 10.2.1.2 host 10.4.1.1 eq telnet

access-list GOOGLE extended deny tcp host 10.3.1.2 host 10.4.1.1 eq telnet

access-list GOOGLE extended deny tcp host 10.1.1.2 host 10.5.1.1 eq telnet

access-list GOOGLE extended deny tcp host 10.3.1.2 host 10.5.1.1 eq telnet

access-list GOOGLE extended deny tcp host 10.1.1.2 host 10.6.1.1 eq telnet

access-list GOOGLE extended deny tcp host 10.2.1.2 host 10.6.1.1 eq telnet

access-group GOOGLE in interface GOOGLE

access-list GOOGLE extended permit tcp host 10.1.1.2 host 10.4.1.1 eq telnet

Crea una regla en la ACL "GOOGLE" que permite el tráfico TCP del host 10.1.1.2 al host 10.4.1.1, específicamente para el servicio Telnet (puerto 23).

access-list GOOGLE extended permit tcp host 10.2.1.2 host 10.5.1.1 eq telnet

Permite el tráfico TCP de 10.2.1.2 hacia 10.5.1.1 también para el servicio Telnet.

access-list GOOGLE extended permit tcp host 10.3.1.2 host 10.6.1.1 eq telnet

Permite el tráfico Telnet entre los hosts 10.3.1.2 y 10.6.1.1.

access-list GOOGLE extended deny tcp host 10.2.1.2 host 10.4.1.1 eq telnet

Niega el tráfico TCP de 10.2.1.2 hacia 10.4.1.1 para Telnet. Este es el primer filtro de tráfico.

```
access-list GOOGLE extended deny tcp host 10.3.1.2 host 10.4.1.1 eq telnet
```

Niega el tráfico Telnet de 10.3.1.2 hacia 10.4.1.1.

```
access-list GOOGLE extended deny tcp host 10.1.1.2 host 10.5.1.1 eq telnet
```

Niega el tráfico Telnet de 10.1.1.2 hacia 10.5.1.1.

```
access-list GOOGLE extended deny tcp host 10.3.1.2 host 10.5.1.1 eq telnet
```

Niega el tráfico Telnet de 10.3.1.2 hacia 10.5.1.1.

```
access-list GOOGLE extended deny tcp host 10.1.1.2 host 10.6.1.1 eq telnet
```

Niega el tráfico Telnet de 10.1.1.2 hacia 10.6.1.1.

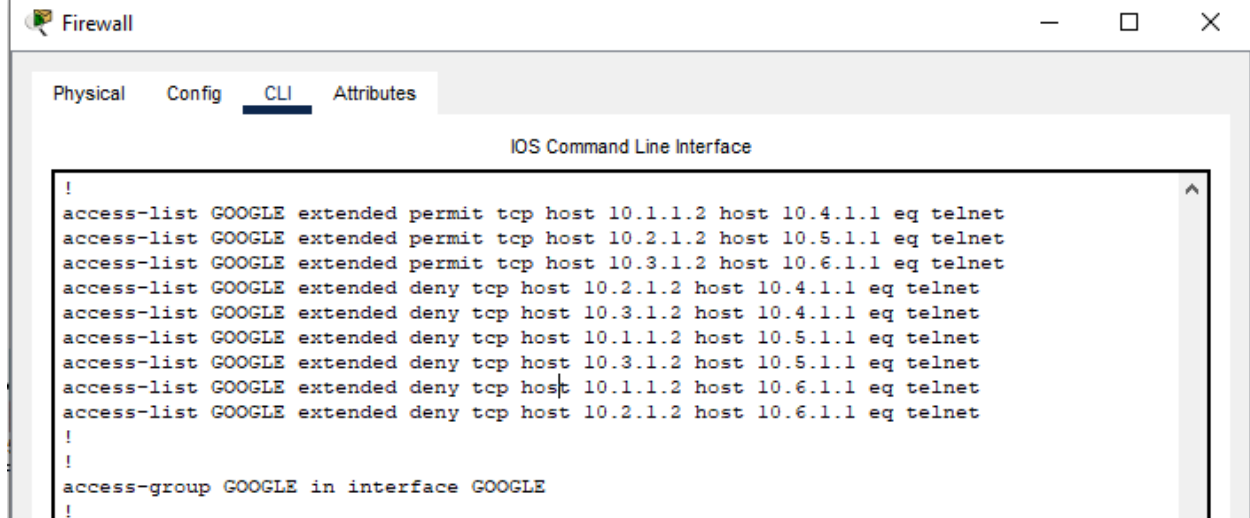
```
access-list GOOGLE extended deny tcp host 10.2.1.2 host 10.6.1.1 eq telnet
```

Niega el tráfico Telnet de 10.2.1.2 hacia 10.6.1.1.

```
access-group GOOGLE in interface GOOGLE
```

Aplica la ACL "GOOGLE" a la interfaz GOOGLE para filtrar el tráfico entrante. Esto significa que el tráfico que ingrese a través de esta interfaz será evaluado según las reglas de la ACL que has definido.

Es muy importante que los denys se pongan al final de los que permiten



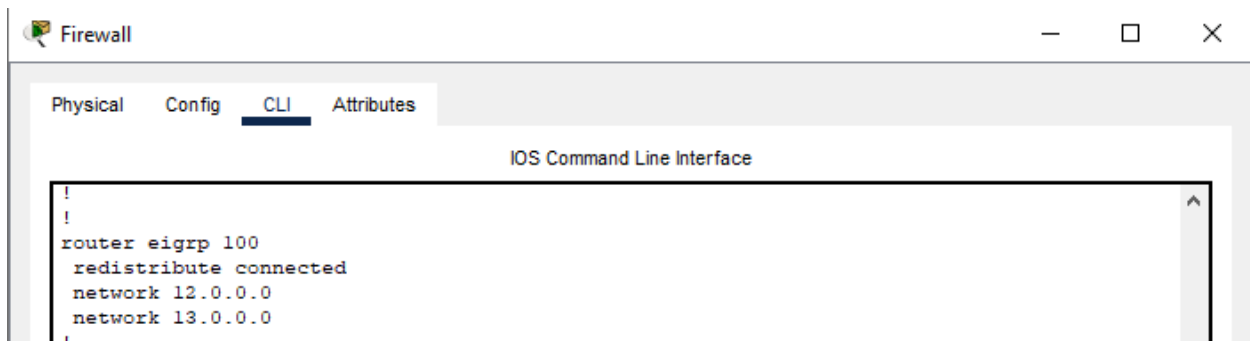
Firewall

Physical Config CLI Attributes

IOS Command Line Interface

```
!  
access-list GOOGLE extended permit tcp host 10.1.1.2 host 10.4.1.1 eq telnet  
access-list GOOGLE extended permit tcp host 10.2.1.2 host 10.5.1.1 eq telnet  
access-list GOOGLE extended permit tcp host 10.3.1.2 host 10.6.1.1 eq telnet  
access-list GOOGLE extended deny tcp host 10.2.1.2 host 10.4.1.1 eq telnet  
access-list GOOGLE extended deny tcp host 10.3.1.2 host 10.4.1.1 eq telnet  
access-list GOOGLE extended deny tcp host 10.1.1.2 host 10.5.1.1 eq telnet  
access-list GOOGLE extended deny tcp host 10.3.1.2 host 10.5.1.1 eq telnet  
access-list GOOGLE extended deny tcp host 10.1.1.2 host 10.6.1.1 eq telnet  
access-list GOOGLE extended deny tcp host 10.2.1.2 host 10.6.1.1 eq telnet  
!  
!  
access-group GOOGLE in interface GOOGLE  
!
```

Por ultimo las conexiones eigrp



Firewall

Physical Config CLI Attributes

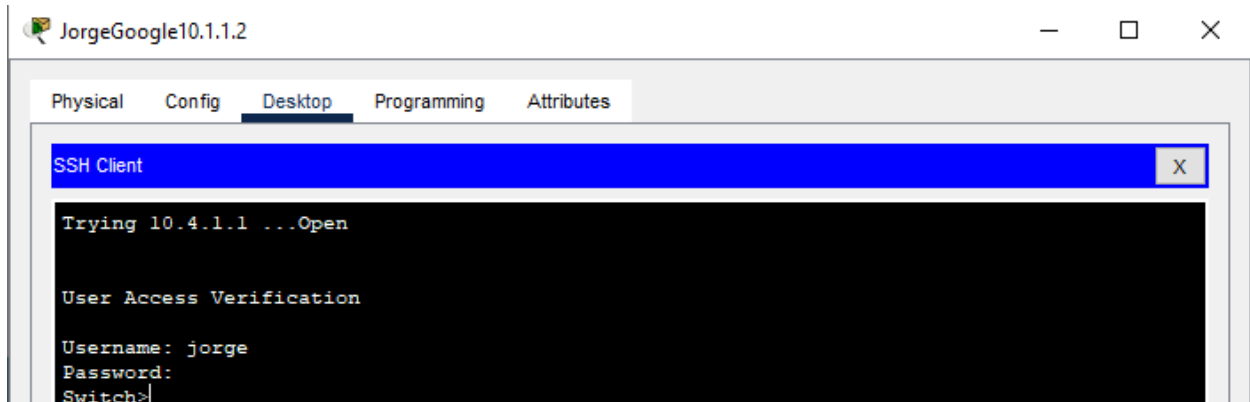
IOS Command Line Interface

```
!  
!  
router eigrp 100  
 redistribute connected  
 network 12.0.0.0  
 network 13.0.0.0  
!
```

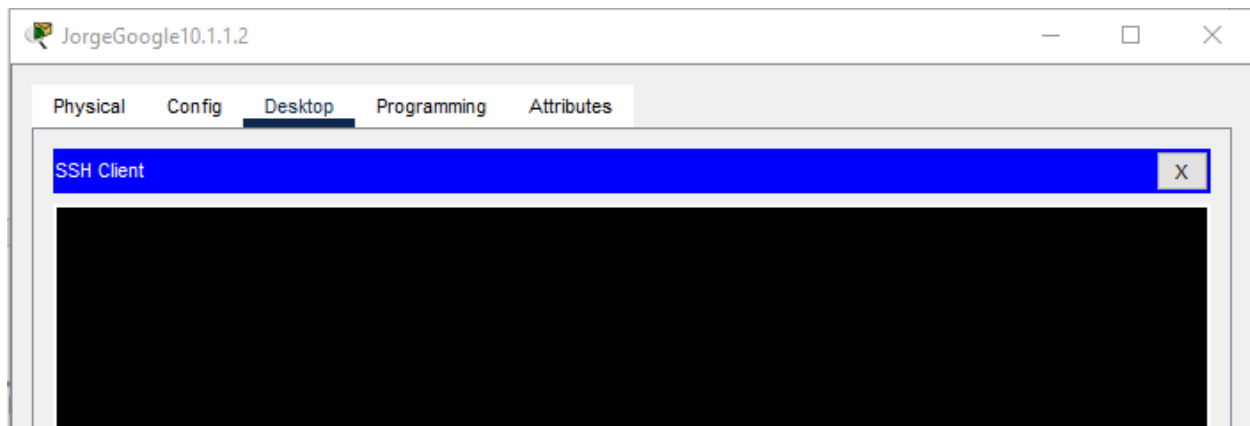
Pings:

PC1 GOOGLE

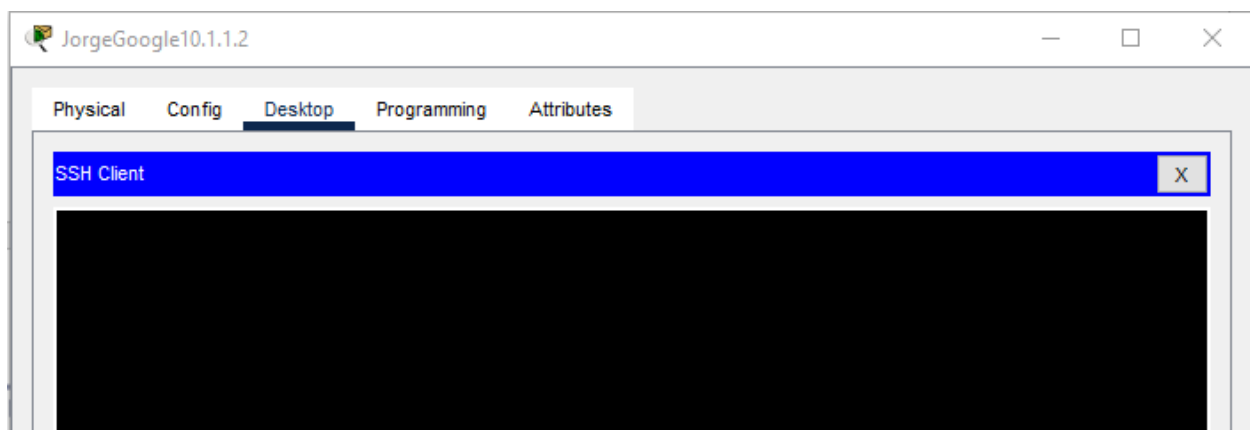
10.4.1.1



10.5.1.1

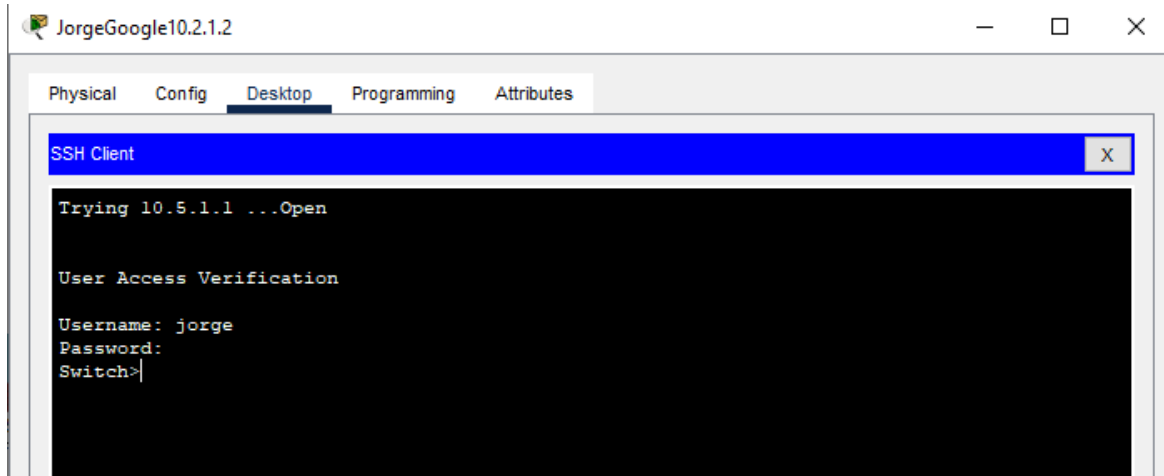


10.6.1.1

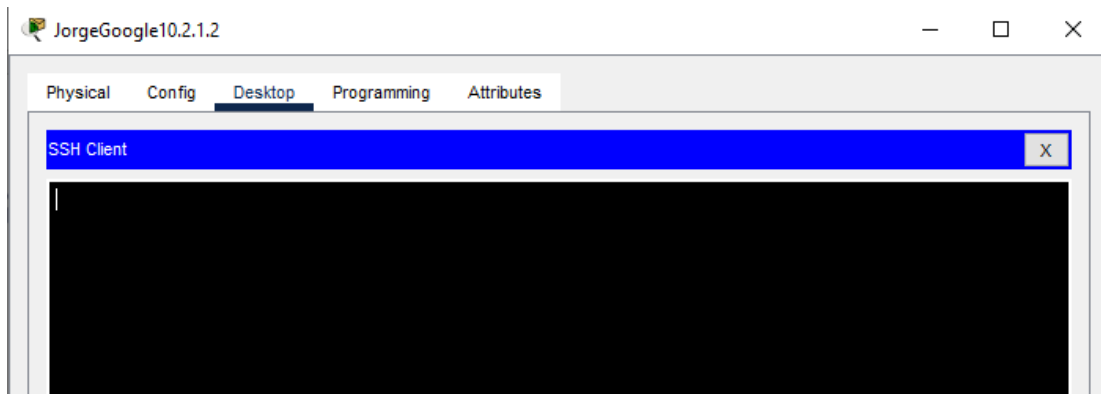


PC2 GOOGLE

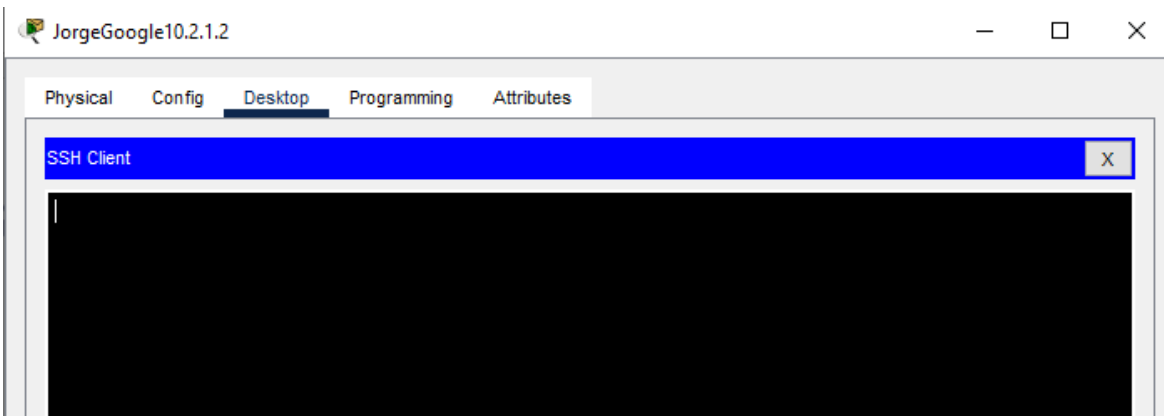
10.5.1.1



10.4.1.1

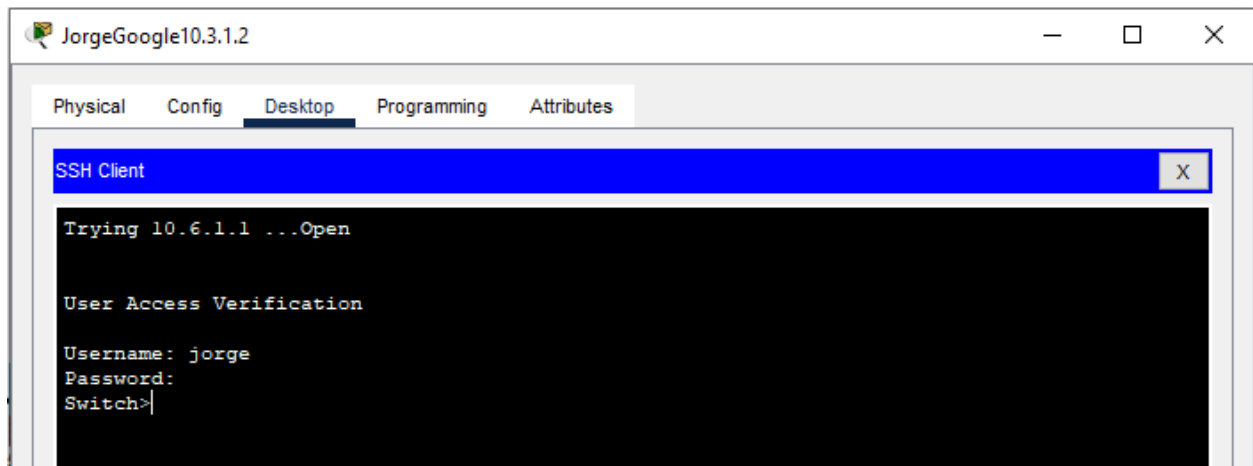


10.6.1.1

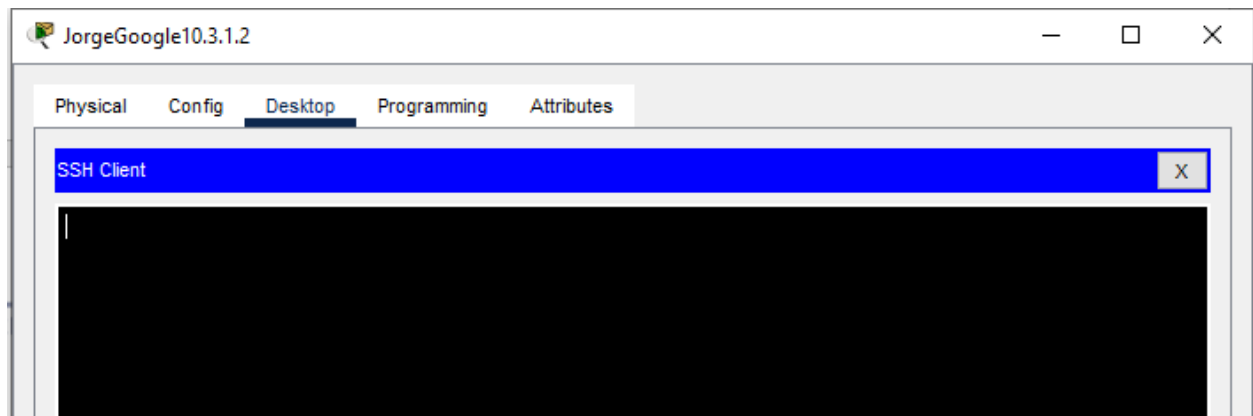


PC3 GOOGLE

10.6.1.1



10.4.1.1



10.5.1.1

