



Alertas en una red de datos  
basada en Netconf-Yang  
Grupo N°6

- Ayala Quinde William
- Jordán Freire Joel
- Parrales Quijije Jahir
- Salazar Castro José

Administración de Sistemas y  
Servicios de Red

2021 – 1S

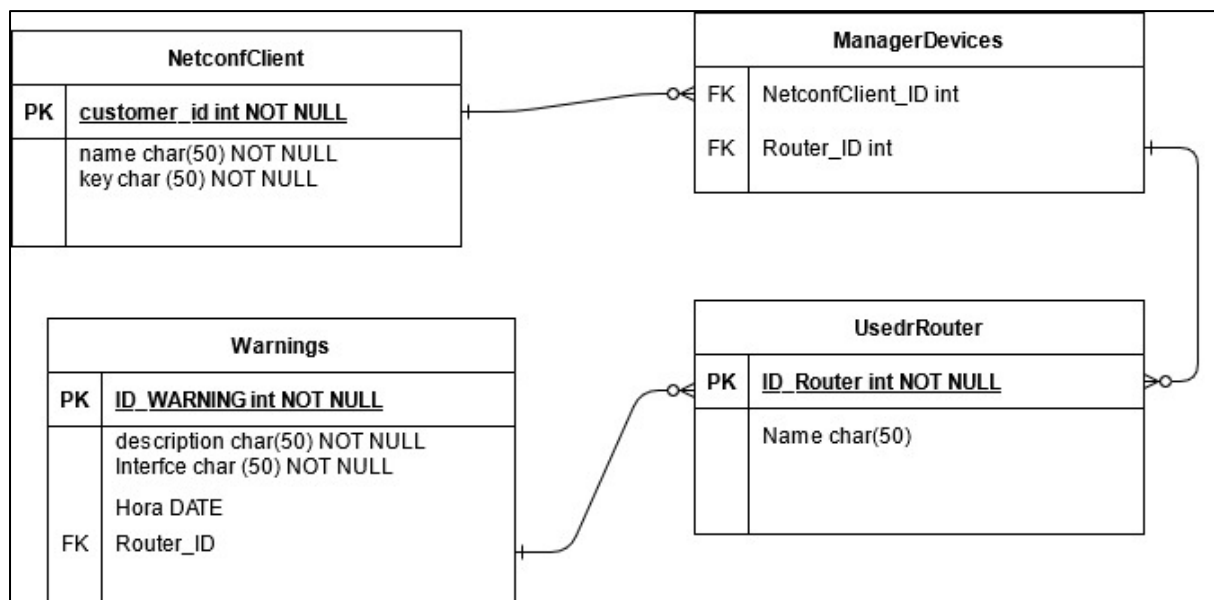
## INTRODUCCIÓN

Mediante el proyecto se busca en desarrollar un sitio web que permita recibir alertas de tráfico en su red de datos, mediante la configuración del protocolo NETCONF-YANG en el Router Cisco XR. Gracias a este protocolo es posible visualizar el tráfico generado en las redes al momento de enviar/recibir alguna trama, además permite realizar configuraciones en los dispositivos que integren la red.

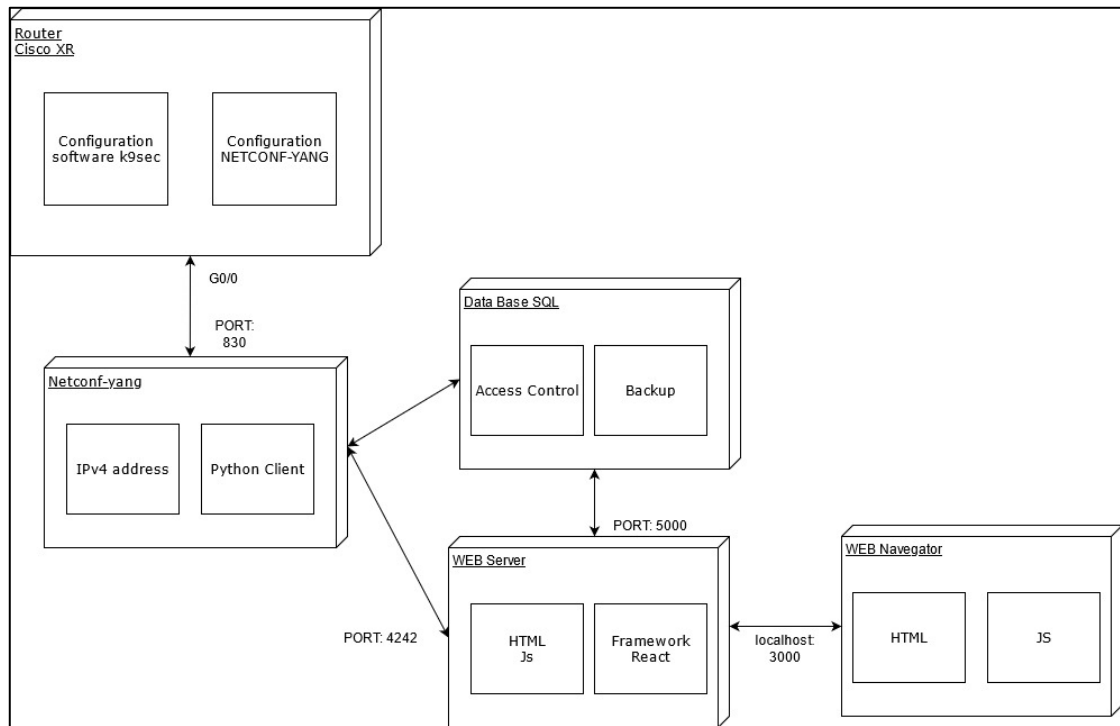
Otra característica de NETCONF-YANG es que nos permite obtener los “loggings” (inicios de sesión) al sistema y lo principal para la realización de este proyecto es la generación de alertas y/o alarmas debido a comportamientos en la red del router.

Debido a la dificultad que representa la configuración del protocolo NETCONFIG, debemos considerar la instalación del paquete de software k9sec pie, el agente NETCONF que habilita la conexión SSH en el router permitiendo que envíe y reciba consultas a través del YANG (lenguaje de consultas) y el cliente NETCONF quien se encarga de recibir las alertas generadas en la red.

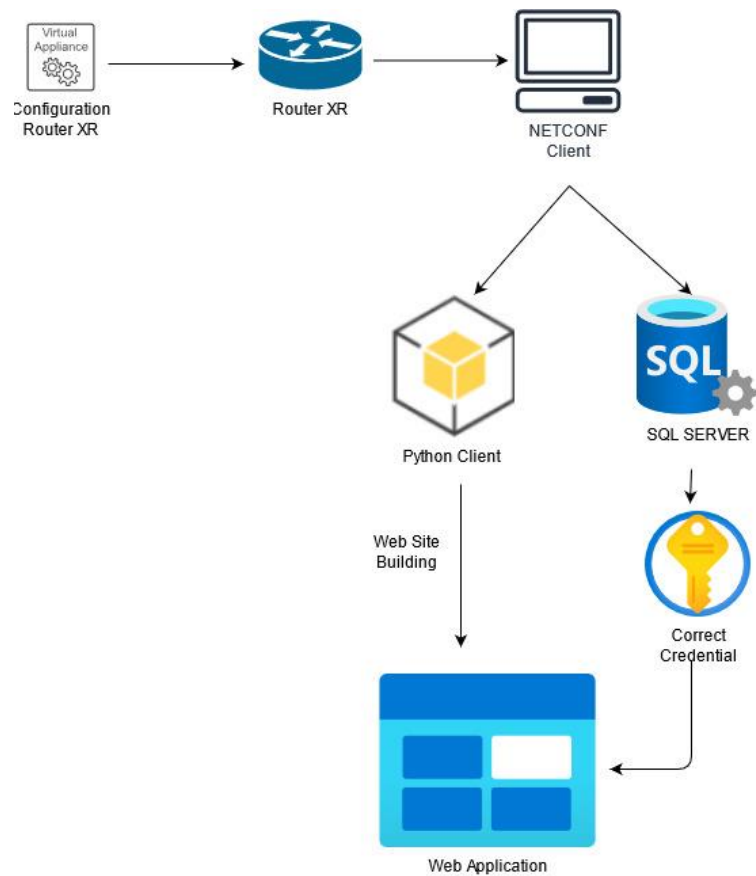
## DIAGRAMA DE ENTIDAD-RELACIÓN



## DIAGRAMA DE DESPLIEGUE

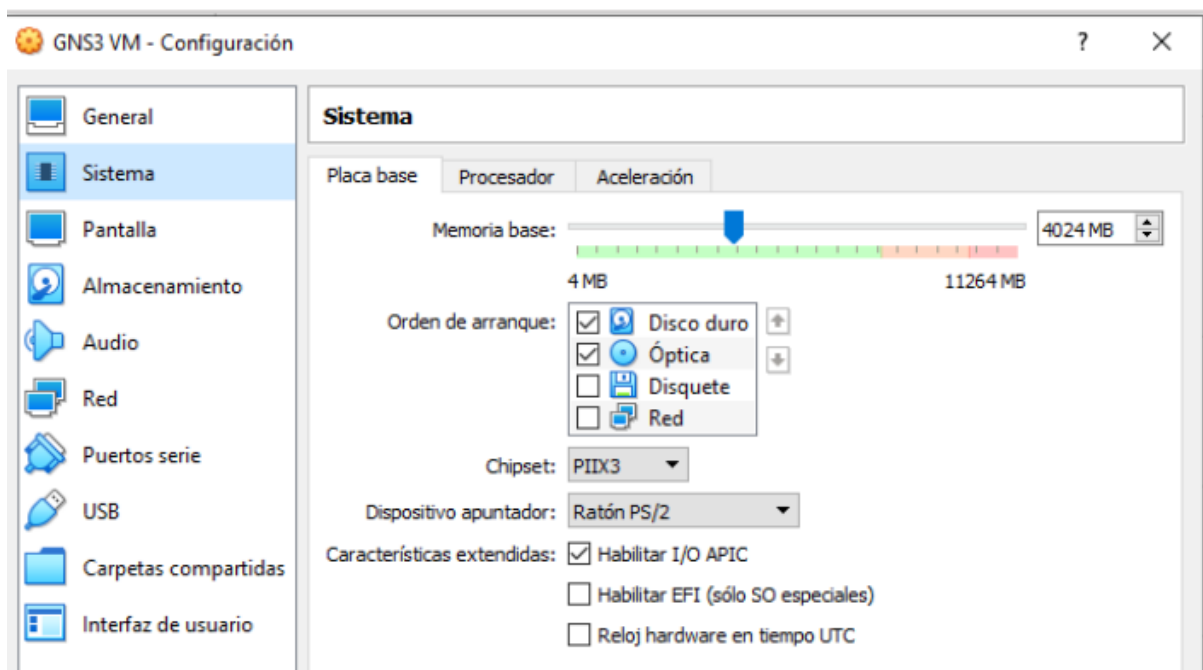


## DIAGRAMA DE PROYECTO



## TOPOLOGIA DE LA RED

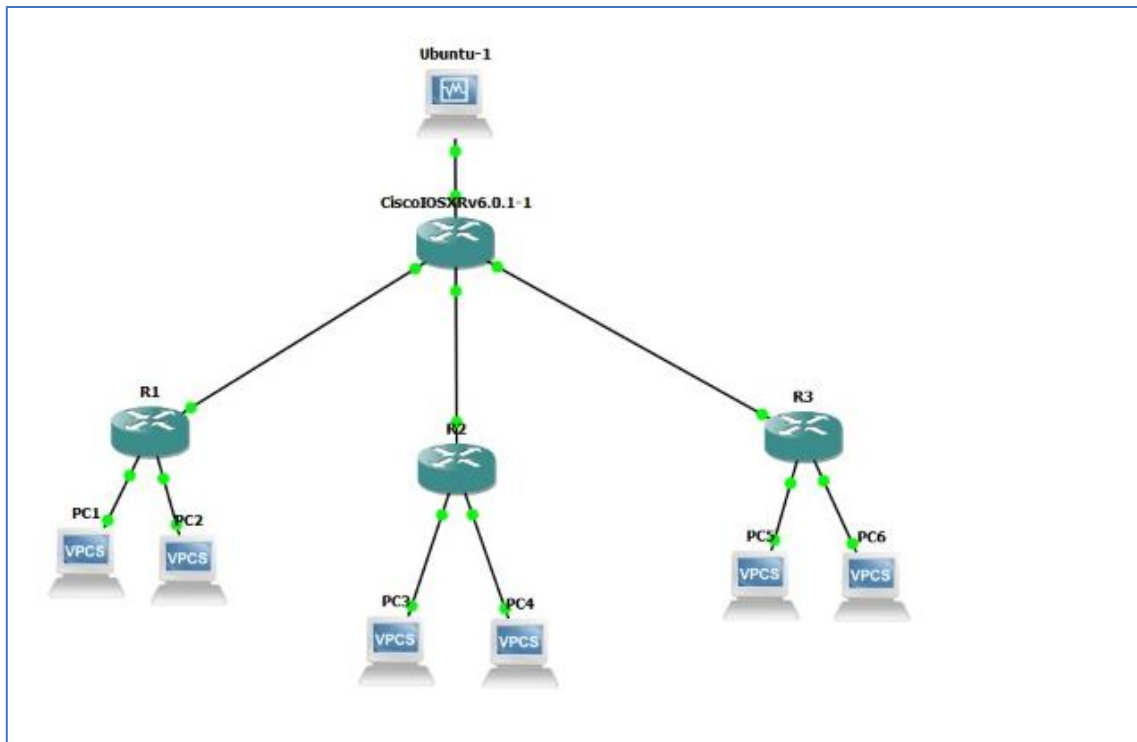
Para la implementación de la topología de red, primero debemos preparar las maquinas virtuales para simulación, para el caso de GNS3 VM se puede descargar en el enlace <https://www.gns3.com/software/download-vm>. Y dentro de las configuraciones de la maquina virtual asignar 2 núcleos de procesamiento, 1024 MB de memoria RAM y un adaptador de red con el fin de instalar el sistema operativo de Ubuntu 20.04



Para este proyecto haremos uso del router XR, para eso es necesario descargar el modelo "Cisco Router XRv6.1.3" El cual requiere 3Gb de RAM que serán tomadas desde GNS3 VM, para ello se necesitan dos ficheros los cuales son:

- Cisco-iosxvr.gns3a, el cual contiene la configuración y diseño del template del router especificado
- Iosxrve-k9-demo-6.1.3.qcow2, la cual es una imagen de prueba del modelo del router Cisco

Una ves instalado la imagen del router Cisco XR, se procede a diseñar la topología de red necesaria, para la realización de este proyecto.



*Ilustración 1 Topología de red*

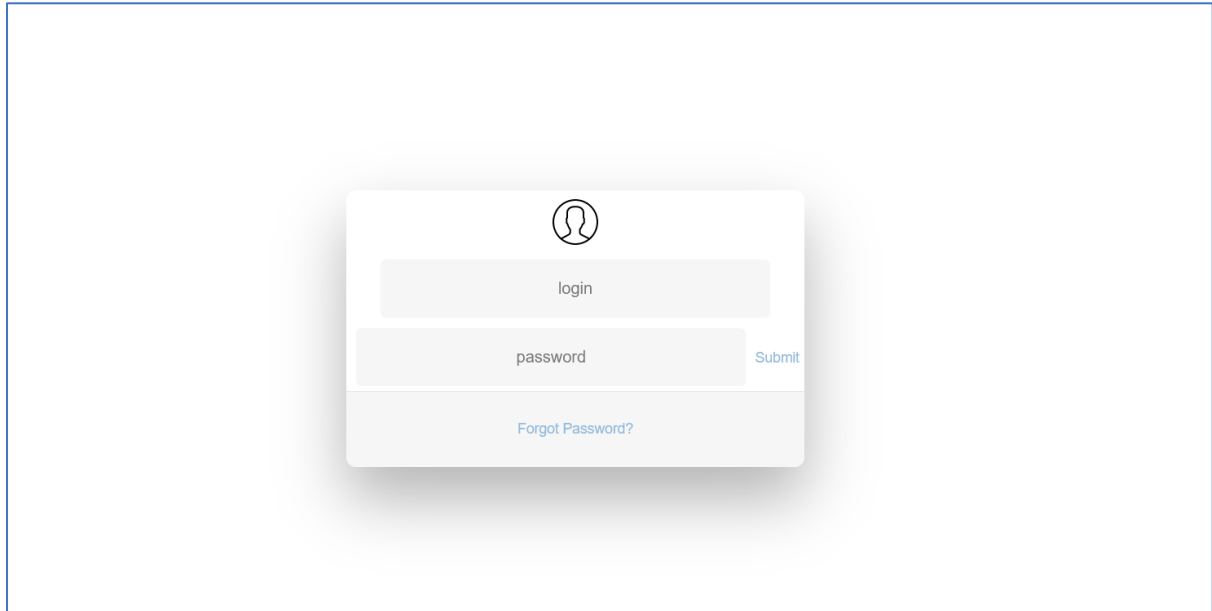
El diseño de la red es a criterio personal, con la condición de que exista al menos un router Cisco XR en esta. Para este caso, se diseñó una red sencilla con el fin de agilizar el proceso de configuración y monitoreo.

## DIRECCIONAMIENTO IPv4

Dispositivo	Interfaz	IPv4	Default Gateway
<b>Router XR</b>	G0/0/0	172.168.10.2 /28	N/A
	G0/0/1	172.168.10.3 /28	N/A
	G0/0/2	172.168.10.4 /28	N/A
	G0/0/3	172.168.10.5 /28	N/A
<b>R1</b>	F0/0	172.168.10.6 /28	N/A
<b>R2</b>	F0/0	172.168.10.7 /28	N/A
<b>R3</b>	F0/0	172.168.10.8 /28	N/A
<b>PC1</b>	E0/0	172.168.20.1 /24	172.168.10.1
<b>PC2</b>	E0/0	172.168.30.1 /24	172.168.10.1

<b>PC3</b>	E0/0	172.168.40.1 /24	172.168.10.1
<b>Ubuntu VM</b>	E0/0	172.168.10.10 /28	172.168.10.1

## RESULTADOS



*Ilustración 2 Inicio de sesión para monitoreo*

En la interfaz LOGIN, se permite que los gestores designados puedan ingresar al sistema de monitoreo de las interfaces conectadas en los routers configurados anteriormente.

*Ilustración 3 Información del sistema*

En la interfaz SYSTEM, se observa la información básica del sistema operativo del router y demás características.

Interfaces						
Interfaces Name	Physic Address	Data In	Data Out	Status	Last Contact	
Interface 1	address 1	1020 bits	500 bits	<span style="color: green;">●</span>	8/15/2021	
Interface 2	address 2	2030 bits	400 bits	<span style="color: red;">●</span>	8/15/2021	
Interface 3	address 3	3010 bits	300 bits	<span style="color: green;">●</span>	8/15/2021	

*Ilustración 4 Interfaces conectadas*

En la interfaz INTERFACES, se observa la información de las interfaces del router, tales como dirección física, cantidad de datos, estado

Addressing						
Interfaces Name	Ipv4 Address	Ipv4 Prefix	Ipv6 Address	Ipv6 Prefix	Status	
Address 1	192.168.11	/24	-	-	<span style="color: green;">●</span>	
Address 2	-	-	20:01:0d:b8:ac:00:30:00:00:00:00:00:00:02	/64	<span style="color: green;">●</span>	
Address 3	-	-	-	-	<span style="color: red;">●</span>	

*Ilustración 5 Direccionamiento IPv4 e IPv6*

En la interfaz ADDRESSING, se observa la tabla de direccionamiento, estado administrativo.

Routing Table			
Protocol	Interface	Next Hop	Time Up
OSPF	G0/0/0/0	20:01:0d:b8:ac:00:30:00:00:00:00:00:00:02/64	12:05:03
OSPF	G0/0/0/1	20:01:0d:b8:ac:00:30:00:00:00:00:00:00:02/128	12:05:03
Local	G0/0/0/4	20:01:0d:b8:ac:00:30:00:00:00:00:00:00:02/64	12:05:03

*Ilustración 6 Protocolos usados para conectar los routers*

En la interfaz ROUTING TABLE, se observa la información sobre el enrutamiento establecido en cada interfaz conectada en el router.

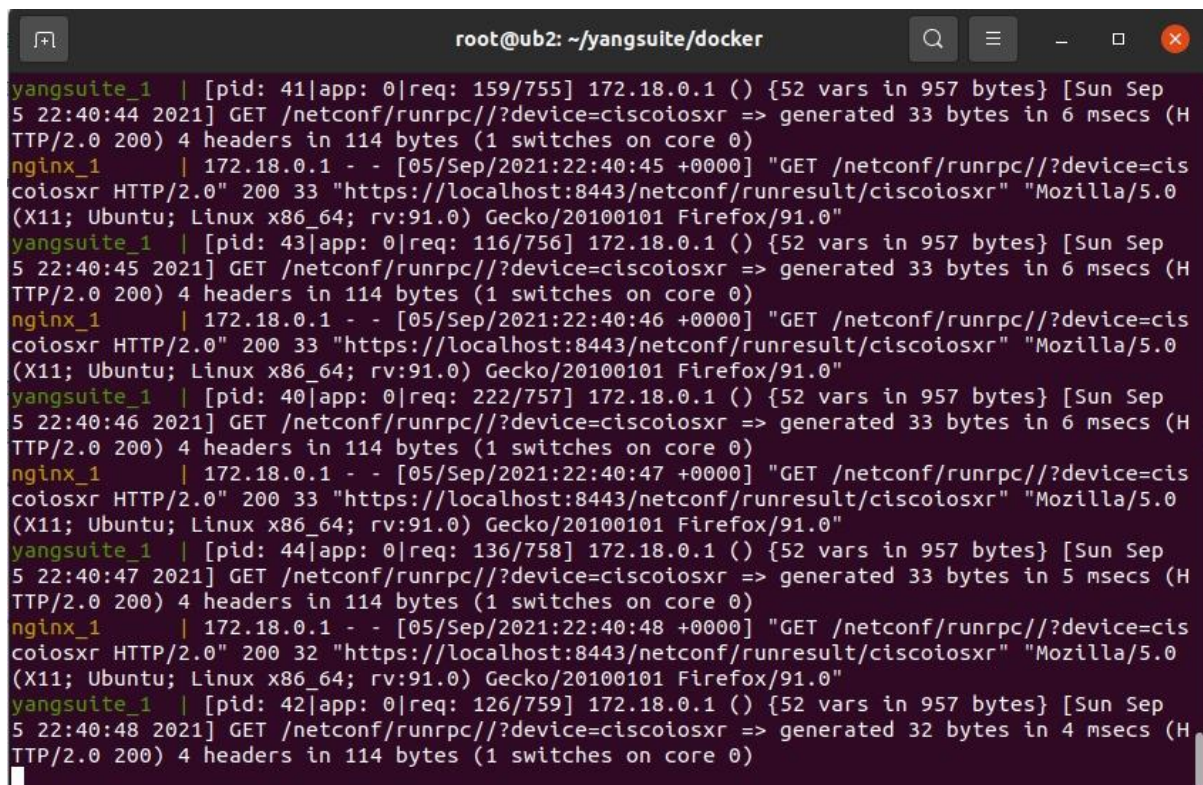
## Alarm Logs

Event ID	Interface	Event Description
4	GigabitEthernet0/0/0	Alta Actividad
3	GigabitEthernet0/0/4	Alta Actividad
2	GigabitEthernet0/0/0	Baja Actividad

Ilustración 7 Notificación de actividades no normales en las interfaces

En la interfaz LOGS, notifica las alarmas más recientes.

## PRUEBA DE CONECTIVIDAD



```
root@ub2: ~/yangsuite/docker

yangsuite_1 | [pid: 41|app: 0|req: 159/755] 172.18.0.1 () {52 vars in 957 bytes} [Sun Sep 5 22:40:44 2021] GET /netconf/runrpc/?device=ciscoiosxr => generated 33 bytes in 6 msecs (HTTP/2.0 200) 4 headers in 114 bytes (1 switches on core 0)
nginx_1      | 172.18.0.1 - - [05/Sep/2021:22:40:45 +0000] "GET /netconf/runrpc/?device=ciscoiosxr HTTP/2.0" 200 33 "https://localhost:8443/netconf/runresult/ciscoiosxr" "Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:91.0) Gecko/20100101 Firefox/91.0"
yangsuite_1 | [pid: 43|app: 0|req: 116/756] 172.18.0.1 () {52 vars in 957 bytes} [Sun Sep 5 22:40:45 2021] GET /netconf/runrpc/?device=ciscoiosxr => generated 33 bytes in 6 msecs (HTTP/2.0 200) 4 headers in 114 bytes (1 switches on core 0)
nginx_1      | 172.18.0.1 - - [05/Sep/2021:22:40:46 +0000] "GET /netconf/runrpc/?device=ciscoiosxr HTTP/2.0" 200 33 "https://localhost:8443/netconf/runresult/ciscoiosxr" "Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:91.0) Gecko/20100101 Firefox/91.0"
yangsuite_1 | [pid: 40|app: 0|req: 222/757] 172.18.0.1 () {52 vars in 957 bytes} [Sun Sep 5 22:40:46 2021] GET /netconf/runrpc/?device=ciscoiosxr => generated 33 bytes in 6 msecs (HTTP/2.0 200) 4 headers in 114 bytes (1 switches on core 0)
nginx_1      | 172.18.0.1 - - [05/Sep/2021:22:40:47 +0000] "GET /netconf/runrpc/?device=ciscoiosxr HTTP/2.0" 200 33 "https://localhost:8443/netconf/runresult/ciscoiosxr" "Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:91.0) Gecko/20100101 Firefox/91.0"
yangsuite_1 | [pid: 44|app: 0|req: 136/758] 172.18.0.1 () {52 vars in 957 bytes} [Sun Sep 5 22:40:47 2021] GET /netconf/runrpc/?device=ciscoiosxr => generated 33 bytes in 5 msecs (HTTP/2.0 200) 4 headers in 114 bytes (1 switches on core 0)
nginx_1      | 172.18.0.1 - - [05/Sep/2021:22:40:48 +0000] "GET /netconf/runrpc/?device=ciscoiosxr HTTP/2.0" 200 32 "https://localhost:8443/netconf/runresult/ciscoiosxr" "Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:91.0) Gecko/20100101 Firefox/91.0"
yangsuite_1 | [pid: 42|app: 0|req: 126/759] 172.18.0.1 () {52 vars in 957 bytes} [Sun Sep 5 22:40:48 2021] GET /netconf/runrpc/?device=ciscoiosxr => generated 32 bytes in 4 msecs (HTTP/2.0 200) 4 headers in 114 bytes (1 switches on core 0)
```

Ilustración 8 Docker en funcionamiento



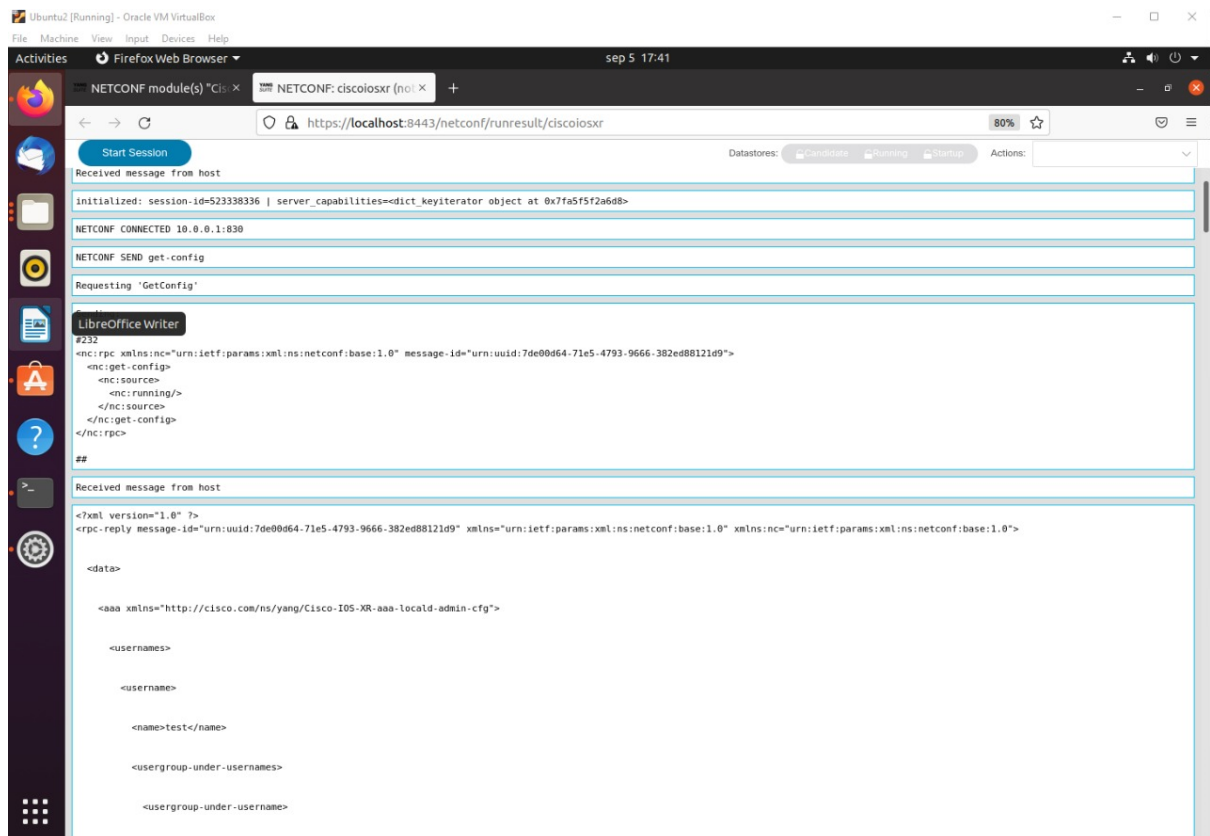


Ilustración 9 aplicación con respuesta de configuración del router

## PRESUPUESTO

<i>Producto/Servicio</i>	<b>Costo Mensual/Unitario</b>	<b>Costo Final</b>
<i>Cisco IOS XR</i>	\$10000	\$10000
<i>Hosting web</i>	\$65	\$780
<i>Amazon RDBS</i>	\$20	\$240

## CONCLUSIONES

- Uno de los aspectos más importantes a considerar en el desarrollo de aplicaciones web, es el tema de la seguridad de los datos que circulan a través del sistema.
- Es necesario que el sistema realizado pueda ser escalable en su momento, es decir, que permita la conexión con otras redes al momento de querer expandirse.
- El uso de diferentes protocolos permite el desarrollo de aplicaciones web que permiten monitorear el tráfico a través de la red usada de una manera sencilla y amigable con el

usuario final, pues en caso contrario, es necesario que el administrador de la red conozca el funcionamiento de cada componente usado, cada protocolo, y cada configuración realizada en los dispositivos usados.

## REFERENCIAS

C. (2021, 6 julio). Configuración de Docker en Ubuntu. Github.

[https://github.com/CiscoDevNet/yangsuite/blob/main/imgs/git\\_clone\\_setup.png](https://github.com/CiscoDevNet/yangsuite/blob/main/imgs/git_clone_setup.png)

C. (2018, 4 octubre). *Configuring Network Configuration Protocol*. Cisco.

[https://www.cisco.com/c/en/us/td/docs/routers/crs/software/crs\\_r5-](https://www.cisco.com/c/en/us/td/docs/routers/crs/software/crs_r5-)

[3/security/configuration/guide/b-syssec-cg53x-](https://www.cisco.com/c/en/us/td/docs/routers/crs/software/crs_r5-3/security/configuration/guide/b-syssec-cg53x-)

[crs/Implementing\\_the\\_Network\\_Configuration\\_Protocol.html#concept\\_ED92A54FE](https://www.cisco.com/c/en/us/td/docs/routers/crs/software/crs_r5-3/security/configuration/guide/b-syssec-cg53x-crs/Implementing_the_Network_Configuration_Protocol.html#concept_ED92A54FE)

[A6143ED8B27ED8F94A771FA](https://www.cisco.com/c/en/us/td/docs/routers/crs/software/crs_r5-3/security/configuration/guide/b-syssec-cg53x-crs/Implementing_the_Network_Configuration_Protocol.html#concept_ED92A54FE)

C. (2020, 9 mayo). *Configure NETCONF/YANG and Validate Example for Cisco IOS XE*

*16.x Platforms*. Cisco. <https://www.cisco.com/c/en/us/support/docs/storage->

[networking/management/200933-YANG-NETCONF-Configuration-Validation.html](https://www.cisco.com/c/en/us/support/docs/storage-networking/management/200933-YANG-NETCONF-Configuration-Validation.html)

Hosting web: <https://proyecto-monitoreo-espol.herokuapp.com/>

Video explicativo: <https://www.youtube.com/watch?v=Ckujd4dE2ig>

Repositorio: <https://github.com/wijoayal/assr/>