



# Tecnológico de Monterrey

## **Network Ready for Use (NRFU)**

Cliente: Ciscorporation

22 de noviembre de 2019

## **Introducción**

El propósito de este documento es definir y registrar las acciones específicas que son indispensables para el uso y prueba de la red cliente, así como para declarar que está lista para ser utilizada. Las pruebas del Network Ready For Use (NRFU) demostrarán que todo el equipamiento de la red ha sido correctamente instalado y configurado, y que la red podrá operar de forma que permita que el cliente la reconozca como un sistema robusto y funcional, que cuente con la conectividad requerida para soportar el tráfico esperado.

Aquí se tendrá documentado a detalle el cómo se ejecutaron las pruebas cruciales para asegurar que la red pueda funcionar de la forma requerida con base en los requerimientos definidos.

## Escenarios de comprobación de Proof of Concept y memoria técnica (Evidencias)

### 1.1 Tablas de direccionamiento

Se recopilaron las direcciones de cada LAN y WAN usadas en la red, así como las de los dispositivos de red y los servidores de las LAN's correspondientes, en un documento técnico utilizado para la instalación de la red.

## Examen Segundo Parcial Redes

Dirección general de la Compañía 172.4.0.0/16

172.4.0.0/24 Servers

Servidor Linux DHCP/DNS WWW

172.4.0.1/24 Linux ← Hosts routers

MPIS : 192.168.0.8/29  
192.168.0.9-14

### Manzanillo

→ Vlan 11 : 250 hosts 172.4.11.0/24 Fa0/2

→ Vlan 12 : 250 hosts 172.4.12.0/24 Fa0/3

Vlan 18 : Admin switches 172.4.18.0/24

12/14 Manzanillo - Colima: 192.168.0.0/29 Fa0/4

→ Puerto Manzanillo: 192.168.0.1/29 - 255.255...258

### Colima

→ Vlan 21: 250 hosts 172.4.21.0/24 Fa0/2

→ Vlan 22: 250 hosts 172.4.22.0/24 Fa0/3

Vlan 28: Admin switches 172.4.28.0/24

24/21 Colima - Manzanillo: 192.168.0.2/29 Fa0/4

Puerto Colima →

→  
Continúa

MPIS Colima: 192.168.0.9/29 Vlan 25 Fa0/5  
Internet Colima, Vlan 26: 10.40.72.20/24 Fa0/6

### Guadalajara

→ Vlan 31 500 hosts: 172.4.30.0/23 Fa0/2

Vlan 37 Servers: 172.4.0.0/24 - 172.4.0.1/24 Fa0/7

Vlan 36 Internet: 10.40.72.20/24 Fa0/6

Vlan 25 MPIS: 192.168.0.10/29 Fa0/5

FTP server 1548 log: 172.4.31.1/23 → está en

Vlan admin switches 38: 172.4.38.0/24

### Monterrey

→ Vlan 41 500 hosts: 172.4.40.0/23 Fa0/2

→ Vlan 43 500 hosts: 172.4.42.0/23 Fa0/3 Access P

Vlan 25 MPIS: 192.168.0.11/29 Fa0/5

Vlan 46 Internet: 10.40.72.20/24 Fa0/6

Vlan 48 Admin-sw: 172.4.48.0/24

Figura 1. Tablas de direccionamiento con direcciones de dispositivos y servidores.

## 1.2 Conectividad entre LAN's y WAN's a través de VLAN's

Se probó la conectividad a través de toda la red gracias a dos pruebas por parte de un técnico. La prueba se llevó a cabo posicionando un dispositivo terminal en una localidad, el cual probó la conectividad con las otras dos localidades. La prueba fue exitosa.

```
[Luiss-MacBook-Pro:~ luisdelgado$ ping 172.4.31.1
PING 172.4.31.1 (172.4.31.1): 56 data bytes
64 bytes from 172.4.31.1: icmp_seq=0 ttl=126 time=1.823 ms
64 bytes from 172.4.31.1: icmp_seq=1 ttl=126 time=2.395 ms
64 bytes from 172.4.31.1: icmp_seq=2 ttl=126 time=1.940 ms
^C
--- 172.4.31.1 ping statistics ---
3 packets transmitted, 3 packets received, 0.0% packet loss
```

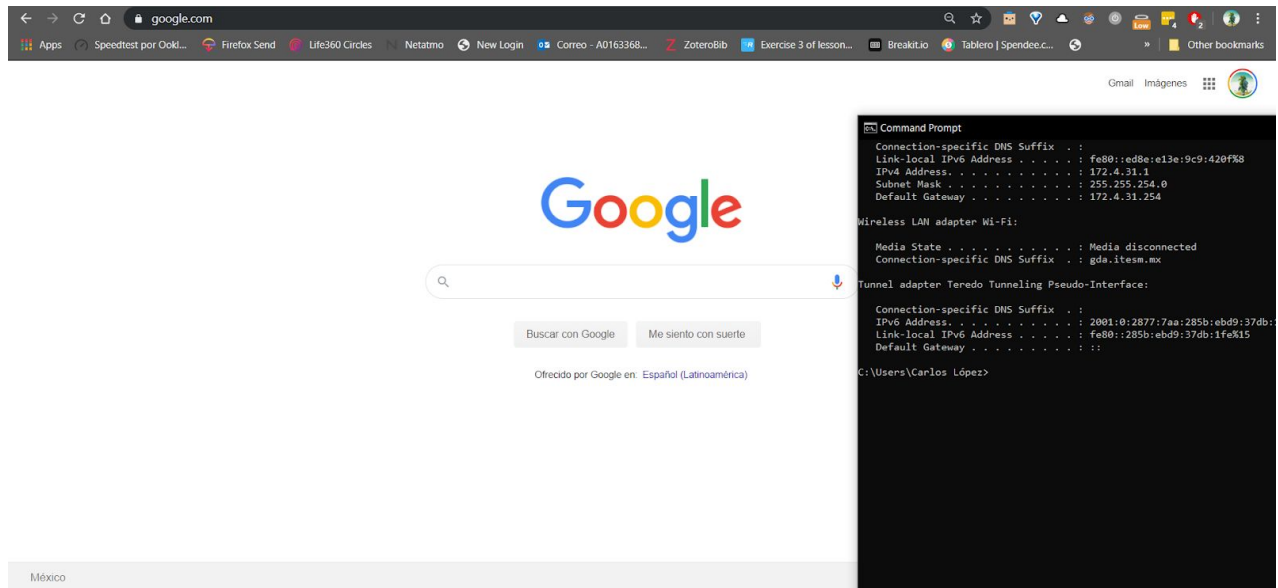
**Figura 2. Conectividad mediante pings exitosa, de Colima a Guadalajara.**

```
[Luiss-MacBook-Pro:~ luisdelgado$ ping 172.4.42.1
PING 172.4.42.1 (172.4.42.1): 56 data bytes
64 bytes from 172.4.42.1: icmp_seq=0 ttl=62 time=1.918 ms
64 bytes from 172.4.42.1: icmp_seq=1 ttl=62 time=2.172 ms
64 bytes from 172.4.42.1: icmp_seq=2 ttl=62 time=1.920 ms
^C
--- 172.4.42.1 ping statistics ---
3 packets transmitted, 3 packets received, 0.0% packet loss
```

**Figura 3. Conectividad mediante pings exitosa, de Colima a Monterrey.**

### 1.3. Salida a internet

La conexión a internet se probó al intentar acceder, a través de un navegador web, a la página principal de Google con una dirección IP brindada por el servidor DHCP de la red instalada. La prueba fue exitosa.

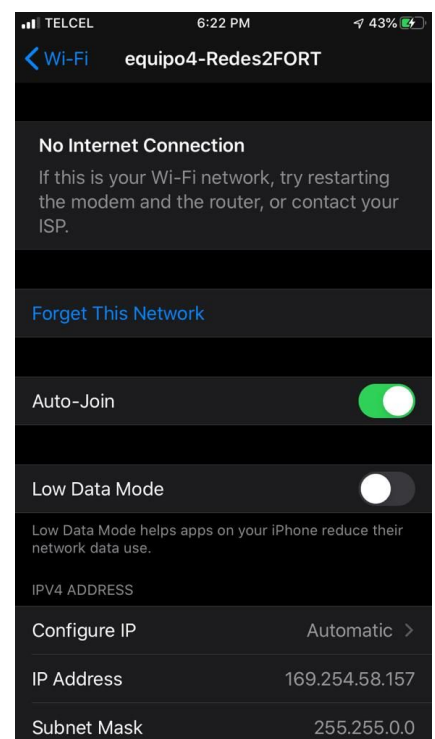


**Figura 4. Acceso exitoso a internet.**

### 1.4. Violación de seguridad en un puerto de acceso con más de una mac address permitida

Gracias a un punto de acceso inalámbrico, se pudieron conectar los suficientes dispositivos para violar el protocolo de seguridad establecido y realizar la prueba.

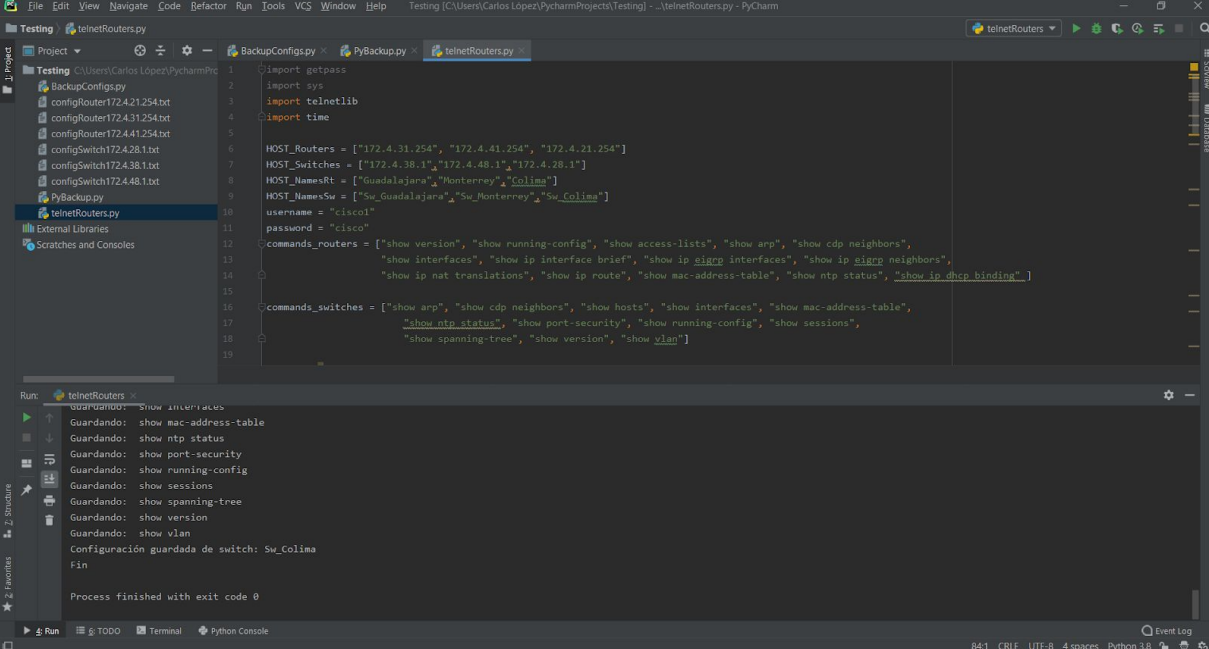
**Figura 5. Sin acceso a internet por violación de protocolo de seguridad.**



## 1.5. Respaldeos de configuraciones automatizado y comprobación de que un equipo administrador puede guardar las configuraciones

Se utilizó scripting con Python para acceder a las configuraciones de todos los dispositivos de red a través de Telnet y respaldarlas de manera segura en un servidor local.

Además, se implementó la funcionalidad de encriptar dichos respaldos con el sistema criptográfico de RSA para agregar una capa de seguridad, y se dejó la posibilidad de ampliar el sistema a futuro con el objetivo de utilizar SSH en vez de Telnet.



```
1 import getpass
2 import sys
3 import telnetlib
4 import time
5
6 HOST_Routers = ["172.4.31.254", "172.4.41.254", "172.4.21.254"]
7 HOST_Switches = ["172.4.38.1", "172.4.48.1", "172.4.28.1"]
8 HOST_NamesRt = ["Guadalajara", "Monterrey", "Colima"]
9 HOST_NamesSw = ["Sw_Guadalajara", "Sw_Monterrey", "Sw_Colima"]
10 username = "cisco"
11 password = "cisco"
12
13 commands_routers = ["show version", "show running-config", "show access-lists", "show arp", "show cdp neighbors",
14                     "show interfaces", "show ip interface brief", "show ip @iprc interfaces", "show ip @iprc neighbors",
15                     "show ip nat translations", "show ip route", "show mac-address-table", "show ntp status", "show ip dhcp binding"]
16
17 commands_switches = ["show arp", "show cdp neighbors", "show hosts", "show interfaces", "show mac-address-table",
18                     "show ntp status", "show port-security", "show running-config", "show sessions",
19                     "show spanning-tree", "show version", "show vlan"]
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
```

Run: telnetRouters.py

```
Guardando: show interfaces
Guardando: show mac-address-table
Guardando: show ntp status
Guardando: show port-security
Guardando: show running-config
Guardando: show sessions
Guardando: show spanning-tree
Guardando: show version
Guardando: show vlan
Configuración guardada de switch: Sw_Colima
Fin
Process finished with exit code 0
```

**Figura 6. Script para respaldo de las configuraciones ejecutado con éxito.**

### 1.6. Acceso al server DHCP Linux entre VLAN's

Era importante para nosotros comprobar conectividad entre nuestras VLAN's, ya que sin conexión entre ellas no podría funcionar la red. Nos aseguramos de mantener conectividad de manera constante, y siempre después de ciclos de trabajo nos aseguramos de tener conectividad.

```
C:\Users\Carlos López>ping 172.4.0.1

Pinging 172.4.0.1 with 32 bytes of data:
Reply from 172.4.0.1: bytes=32 time=1ms TTL=63
Reply from 172.4.0.1: bytes=32 time=1ms TTL=63
Reply from 172.4.0.1: bytes=32 time=2ms TTL=63
Reply from 172.4.0.1: bytes=32 time=2ms TTL=63

Ping statistics for 172.4.0.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 1ms, Maximum = 2ms, Average = 1ms

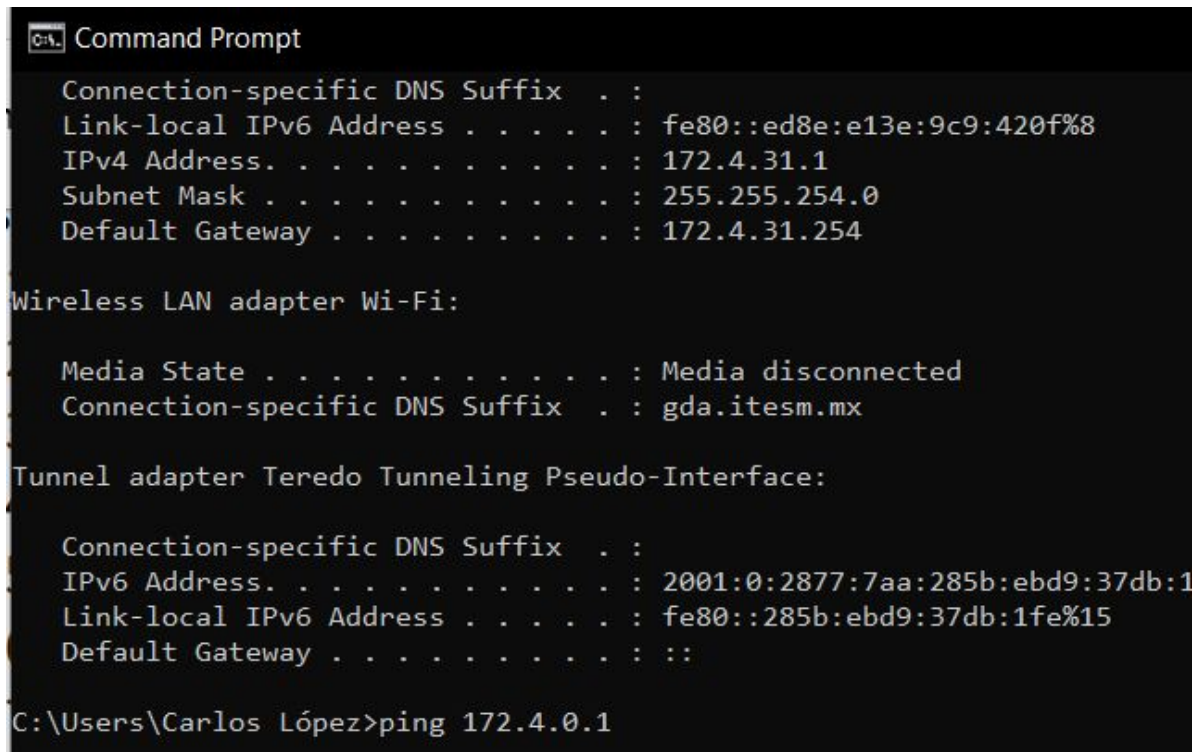
C:\Users\Carlos López>_
```

**Figura 7. Acceso al servidor DHCP mediante ping exitoso.**



## 1.7. Entregar una dirección fija vía DHCP Linux

Con LINUX creamos un servidor DHCP con el cual brindaba direcciones a los equipos que lo solicitaban. Permite excluir también algunas direcciones, esto con la finalidad de apartar direcciones para equipos de uso futuro como impresoras, ventiladores, luces, o equipos electrodomésticos que se usan con mayor frecuencia últimamente.



```
C:\> Command Prompt

Connection-specific DNS Suffix . : 
Link-local IPv6 Address . . . . . : fe80::ed8e:e13e:9c9:420f%8
IPv4 Address. . . . . : 172.4.31.1
Subnet Mask . . . . . : 255.255.254.0
Default Gateway . . . . . : 172.4.31.254

Wireless LAN adapter Wi-Fi:

Media State . . . . . : Media disconnected
Connection-specific DNS Suffix . : gda.itesm.mx

Tunnel adapter Teredo Tunneling Pseudo-Interface:

Connection-specific DNS Suffix . : 
IPv6 Address. . . . . : 2001:0:2877:7aa:285b:ebd9:37db:1
Link-local IPv6 Address . . . . . : fe80::285b:ebd9:37db:1fe%15
Default Gateway . . . . . : ::

C:\Users\Carlos López>ping 172.4.0.1
```

**Figura 8.** Dispositivo en Guadalajara con dirección IP asignada por el servidor DHCP.

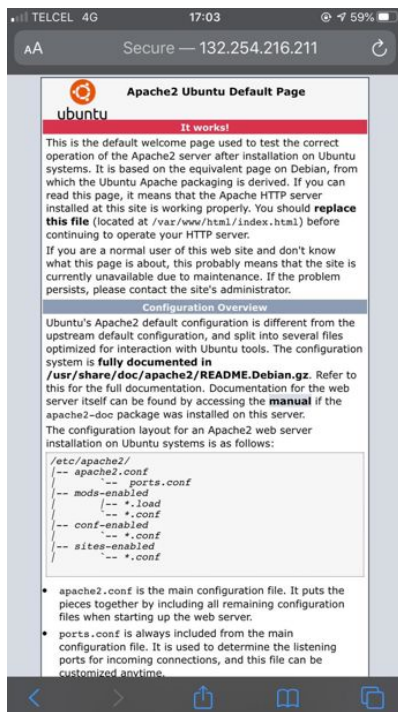
## 1.8. Evidencia de filtros dinámicos

La prueba con filtros dinámicos fue guardada en un documento externo, el cual fue anexado a esta entrega con el motivo de detectar posibles errores y observar el comportamiento de los filtros dinámicos con el paso del tiempo. La salida de la consola se encuentra en el archivo “Evidencia filtros dinámicos y error en conf. clock” en la carpeta de Anexos.



### 1.9. Acceder al servidor apache desde internet

El servidor web fue ejecutado de manera correcta gracias a Apache, herramienta con la cual el cliente será capaz de acceder a los sitios web deseados desde cualquier parte del mundo siempre que cuente con internet. La prueba fue realizada sobre una plataforma móvil con datos celulares para probar la funcionalidad del servicio.



**Figura 9. Servidor apache corriendo con acceso desde internet con datos celulares.**

### 1.10. Video de documentación de la teoría aplicada

El video de documentación se encuentra como parte de los anexos y requisitos del proyecto final. En este video se explica detalladamente el desarrollo del proyecto, desde cableado y funcionalidad de cada equipo, switches y routers, hasta cómo configuramos servidores y demás configuraciones. A continuación se adjunta el link del video en youtube.

<https://www.youtube.com/watch?v=jwrwfl7YtDk&feature=youtu.be>