



INSTITUTO POLITÉCNICO NACIONAL
ESCUELA SUPERIOR DE CÓMPUTO



ADMINISTRACIÓN DE SERVICIOS EN RED

REPORTE

PROYECTO FINAL

Integrantes:

- García Vera Jared Alberto
- Gracia Barajas Karla Alejandra
- Molina Santiago Isaac
- Sánchez Robles Andrea Selene

Profesora: Leticia Henestrosa
Carrasco

Grupo: 4CV13

Fecha de entrega: 20/12/2021

Introducción

La administración de red se define como el proceso de administración de una red de los fallos y el rendimiento, a continuación, se describirán diversos conceptos y protocolos que ayudan a llevar estas tareas a cabo:

OSPF (Open Shortest Path First)

Protocolo de enrutamiento dinámico, que usa un algoritmo de tipo Estado - Enlace.

En esencia este protocolo se basa en:

- Aprender información de enrutamiento sobre las subredes IP de los routers vecinos.
- Anunciar información de enrutamiento sobre subredes IP a los routers vecinos, si existe más de una ruta posible para llegar a una subred, elige la mejor ruta con base en una métrica.
- Si la tipología de la red cambia, por ejemplo, si un enlace falla, reacciona anunciando que algunas rutas han fallado y elige la nueva mejor ruta.

VLSM (Variable Length Subnet Mask)

Para la realización del subneteo se empleó VLSM, siendo este el resultado del proceso por el cual se divide una red o subred en subredes más pequeñas cuyas máscaras son diferentes según se adaptan a las necesidades de hosts por subred, generando una máscara diferente para las distintas subredes de una red.

VLAN (Virtual LAN)

Nos permite crear redes lógicamente independientes dentro de la misma red física, por lo tanto, podremos aislarlas para que solamente tengan conexión a Internet, y denegar el tráfico de una VLAN a otra.

NAT (Network Address Translator)

Se trata de un sistema que se utiliza en las redes bajo el protocolo IP y que nos permite el intercambio de paquetes entre dos redes que tienen asignadas mutuamente direcciones IP incompatibles

ACL extendidas (Access Control List)

La ACL extendida permite o deniega el acceso según la dirección IP de origen, la dirección IP de destino, el tipo de protocolo y los números de puertos. Dado que las ACL extendidas pueden ser muy específicas, tienden a aumentar su tamaño rápidamente.

SNMP (Simple Network Management Protocol)

Es un protocolo de capa de aplicación basado en IP que intercambia información entre una solución de administración de red y cualquier dispositivo habilitado para SNMP.

DNS (Domain Name System)

Es un sistema de nomenclatura jerárquico descentralizado para dispositivos conectados a redes IP como Internet o una red privada. Este sistema asocia información variada con nombres de dominio asignados a cada uno de los participantes.

DHCP (Dynamic Host Configuration Protocol)

Es un protocolo de red de tipo cliente/servidor mediante el cual un servidor DHCP asigna dinámicamente una dirección IP y otros parámetros de configuración de red a cada dispositivo en una red para que puedan comunicarse con otras redes IP.

HTTP (Hypertext Transfer Protocol)

Es un protocolo de la capa de aplicación para la transmisión de documentos hipertexto. Fue diseñado para la comunicación entre los navegadores y servidores web, aunque puede ser utilizado para otros propósitos también.

TFTP (Protocolo de transferencia de archivos trivial)

TFTP es una versión simplificada de FTP (Protocolo de transferencia de archivos). Fue diseñado para ser fácil y sencillo. TFTP omite muchas funciones de autenticación de FTP y se ejecuta en el puerto UDP 69. Como es muy ligero, todavía se utiliza para diferentes propósitos.

TFTP se usa en lugares donde no se necesita mucha seguridad. En su lugar, necesita una forma de cargar y descargar archivos fácilmente desde el servidor. Los dispositivos CISCO utilizan el protocolo TFTP para almacenar archivos de configuración e imágenes CISCO IOS con fines de respaldo. Los protocolos de arranque de red como BOOTP, PXE, etc. utilizan TFTP para arrancar sistemas operativos a través de la red. Los clientes ligeros también utilizan el protocolo TFTP para arrancar sistemas operativos. Muchas placas de circuitos electrónicos y microprocesadores también usan TFTP para descargar firmware en el chip. En general, TFTP tiene muchos usos incluso hoy.

Objetivo

Construir y montar una topología configurada para brindar los servicios de SNMP, DNS, DHCP, HTTP Y TFTP, implementando Azure y el emulador GNS3.

Desarrollo del proyecto

Contexto

Una institución que ofrece 2 niveles educativos:

1. **Nivel básico** (capacidad de 2000 estudiantes)
2. **Nivel medio superior** (capacidad de 2000 estudiantes)
2.1 **Nivel técnico** (capacidad de +2000 estudiantes, ya que personas externas a la institución pueden cursar un nivel técnico en la institución).

Cada nivel conforma una LAN.

Cada nivel educativo está conformado por laboratorios, área administrativa de alumnos y docentes (altas, bajas, registros, calificaciones*, horarios, historiales), así como también del área de becas.

Cada LAN de cada nivel educativo tiene como mínimo 4 subredes.

La institución ofrece al menos los siguientes servicios a la comunidad:

1. Página web donde se consultan calificaciones. www.calificaciones-alumnos.com
2. Página web donde se suben calificaciones www.calificaciones-docentes.com
3. Plataforma web de contenido didáctico para los alumnos. www.material-didactico.com
4. Plataforma web donde se suben los exámenes www.examenes-supervision.com

Y además tienen una LAN dentro de la misma red llamada Administración General, que se encarga de gestionar recursos financieros y administración de personal.

Dentro de esa LAN se encuentran los servidores principales, como los son SNMP, DNS y DHCP, además de contar con los servidores para FTP que ayudan a tener las plataformas web actualizadas, así como el servidor de HTTP.

Dentro de la LAN de Administración General hay 3 subredes y 3 servidores.

La institución cuenta con un ISP.

Accesibilidad entre redes

1. No habrá acceso a la plataforma www.calificaciones-docentes.com desde los equipos que conforman los laboratorios en cualquier nivel.
2. No habrá acceso a la plataforma www.examenes-supervision.com

3. Los equipos de los laboratorios no pueden acceder a los servidores.
4. Todas las LAN podrán salir a internet, pero no podrán acceder a todas las páginas.

Seguridad

Se utilizará PAT con un pool de direcciones para las LAN de los niveles educativos

Se utilizará PAT con una única dirección para el servidor de HTTP.

Diagrama

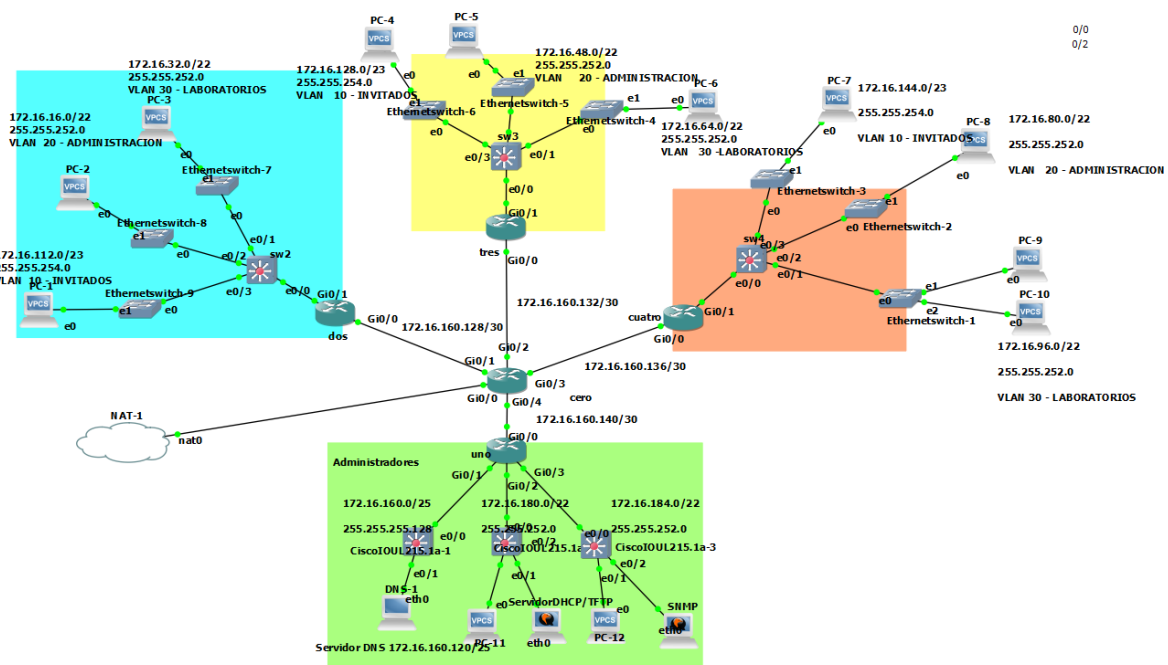


Ilustración 1 Topología

Servicios

SNMP :

1. Nos notificará algún enlace caído y su recuperación.
2. Notificará cuando algún servidor se detenga.
3. Notificará el acceso incorrecto al servidor FTP.

DNS :

1. Nos proporcionará acceso mediante el nombre de dominio de las páginas que la institución ofrece.

DHCP :

1. Nos dará las IPs de todas las LAN de manera dinámica y las IPs de los servidores estarán de manera estática.

HTTP :

1. Para proporcionar a los alumnos y docentes las páginas funcionando.

FTP:

1. Gestión de los documentos de las páginas web.

Protocolo de enrutamiento

- RIPv2 con VLSM

Escenarios

1. Estado de los enlaces caídos y levantados.
2. Respaldo de configuración de los dispositivos.
3. Reporte de los intentos fallidos de conexión de usuarios.
4. Que las páginas web se pueden consultar o no según el área y LAN.
5. Que se puedan consultar, subir, actualizar y eliminar archivos en el servicio FTP.

DNS

Para la implementación del servicio se hizo en una “máquina virtual” sin interfaz gráfica, esto para ahorrar recursos en la máquina principal de Azure. Antes de comenzar a ver la configuración de la máquina virtual debemos primero configurar los routers de la topología para indicarle en donde se encuentra el servidor DNS usando los siguientes comandos:

1. **configure terminal** -> Habilita el modo de configuración del router.
2. **ip domain-lookup** -> Habilita la traducción de nombre a dirección basado en DNS del host.
3. **ip name-server 172.16.160.120** -> Especifica la dirección IP en donde está alojado nuestro servidor DNS, en este caso está en la dirección 172.16.160.120.

Esta configuración debe aplicarse a todos los routers. Una vez terminada esta configuración, se accede a la máquina virtual y se escribe el siguiente comando:

- **nano /etc/network/interfaces** - > Accedemos al archivo de configuración de las interfaces de red de la máquina virtual, descomentamos la parte de asignación estática y comentamos la parte de DHCP. Asignamos la dirección IP escribiendo lo siguiente:
 - ip address 172.16.160.120
 - netmask 255.255.255.128

- gateway 172.16.160.1
- up echo nameserver 192.168.122.1 > /etc/resolv.conf

Esta última instrucción nos sirve para indicar la IP del servidor DNS.

Guardamos el archivo y escribimos en la terminal **service dnsmasq restart** para habilitar el servicio.

Posteriormente, se edita el siguiente archivo de configuración escribiendo el siguiente comando:

- **sudo nano /etc/dnsmasq.conf** -> Sirve para acceder al archivo de configuración del servidor.

Dentro del archivo se escriben las siguientes reglas:

1. **interface=eth0** -> Indicamos el puerto de red por donde se habilitará el servicio.
2. **listen-address=127.0.0.1** -> Indicamos IP por donde se escuchará.

Guardamos el archivo, y escribimos el siguiente comando para especificar los dominios:

- **sudo nano /etc/hosts**
y dentro de él se escribe lo siguiente:
 - 209.165.200.226 mi_escuela.com.lab mi_escuela -> especificamos que "mi_escuela.com.lab" tiene asociada la dirección IP 209.165.200.226.
 - 172.16.160.120 dnserver.com dnserver -> especificamos que "dnserver.com" tiene asociada la dirección IP 172.16.160.120.
 - 74.125.230.82 www.google.es google.es -> especificamos que "www.google.es" tiene asociada la dirección IP 74.125.230.82.

Guardamos el archivo y reiniciamos la red con el comando **sudo /etc/init.d/dnsmasq restart**.

Nota: El archivo **/etc/resolv.conf** es inmutable, es decir, no se puede modificar o borrar aún con permisos root, para poderlo editar se usó el comando **sudo chattr -i /etc/resolv.conf**.

SNMP

Para la implementación de este servicio se hizo directamente en la máquina virtual de ubuntu que montamos, y nos apoyamos de los siguientes comandos desde la terminal de la máquina virtual:

1. **sudo apt-get update** -> actualizar paquetes de ubuntu
2. **sudo apt-get install snmpd** -> instalar SNMP
3. **sudo nano /etc/snmp/snmpd.conf** -> se editara el archivo de configuración del snmp
 - a. Dentro del archivo de configuración se descomentara la línea donde viene "agentAddress udp:161,udp6:[::1]:161", para que de esta manera el servidor escuche todas las direcciones IPv4 y IPv6
4. Se configurara la comunidad a la que queremos que pertenezca cambiando el nombre de rocommunity public por proj_redes3_rw y proj_redes3_ro , guardamos y salimos.
5. **sudo service snmpd restart** -> reiniciamos el servidor SNMP
6. **sudo service snmpd status** -> visualizamos que el server de SNMP se inició de manera correcta

```
11-3 sudo service snmpd status
● snmpd.service - Simple Network Management Protocol (SNMP) Daemon
   Loaded: loaded (/lib/systemd/system/snmpd.service; enabled; vendor preset: enabled)
   Active: active (running) since Tue 2020-07-21 09:33:07 UTC; 7min ago
     Process: 4390 ExecStartPre=/usr/sbin/snmpd -p /var/run/snmpd (code=exited, status=0/SUCCESS)
    Main PID: 4390 (snmpd)
       Tasks: 1 (limit: 2282)
      Memory: 5.6M
    CGroup: /system.slice/snmpd.service
           └─4390 /usr/sbin/snmpd -LOW -u Debian-snmp -g Debian-snmp -I -smux mteTrigger mteTriggerConf -f -p /run/s

Jul 21 09:33:07 unifi systemd[1]: Starting Simple Network Management Protocol (SNMP) Daemon....
Jul 21 09:33:07 unifi systemd[1]: Started Simple Network Management Protocol (SNMP) Daemon..
lines 1-12/12 (END)
```

Ilustración 2 Configuración SNMP

HTTP

Para la implementación de este servicio se apoyó en la documentación en CISCO para la configuración del servicio HTTP

```
cero
*Dec 20 05:30:40.913: %OSPF-5-ADJCHG: Process 1, Nbr 172.16.184.1 on GigabitEthernet0/4 from LOADING to FULL, Loading Done
*Dec 20 05:31:06.529: %LINK-2-INTVULN: In critical region with interrupt level=0, intfc=GigabitEthernet0/4 -Process= "IP Input", ipl= 0, pid= 137
-Traceback= 115004Az 218B85z 13E46Bz 162431Bz 1626FC0z 163EA3Ez 163E40Bz 14D8BDz 14CF84z 15886Az 1597CBz 15975Fz 21DDD3z 19A1258z 19A0AACz 199D4E2z
*Dec 20 05:31:53.733: %PLATFORM-5-SIGNATURE_VERIFIED: Image 'flash0:/vios-adventerprisek9-m' passed code signing verification
*****
* IOSv is strictly limited to use for evaluation, demonstration and IOS *
* education. IOSv is provided as-is and is not supported by Cisco's *
* Technical Advisory Center. Any use or disclosure, in whole or in part, *
* of the IOSv Software or Documentation to any third party for any *
* purposes is expressly prohibited except as otherwise authorized by *
* Cisco in writing. *
*****
Rcero#config t
Enter configuration commands, one per line. End with CNTL/Z.
Rcero(config)#username equipo7 privilege 0 secret escom
Rcero(config)#ip http server
Rcero(config)#exit
Rcero#
*Dec 20 05:44:49.298: %SYS-5-CONFIG_I: Configured from console by console
Rcero#
```

Ilustración 3 Configuración en router para http

En el router se entró en el modo de configuración, posteriormente se creó un usuario con su respectiva contraseña y se le asignó un nivel de privilegio, esto se hace porque al ingresar a la dirección del router desde nuestra máquina virtual nos pedirá un usuario y contraseña para poder entrar.

El nivel de acceso que se le quiere dar al usuario va acorde a la siguiente imagen.

<i>level</i>	Indicates a privilege value from 0 through 15. By default, there are the following three command privilege levels on the router: <ul style="list-style-type: none">0--Includes the disable , enable , exit , help , and logout commands.1--Includes all user-level commands at the router prompt (>).15--Includes all enable-level commands at the router prompt (>).
--------------	--

Para habilitar el servicio de HTTP en el router se empleó el siguiente comando:

ip http server

Una vez que habilitamos el servicio de HTTP en el router, solo queda salir del modo configuración y guardas los cambios en el router.

```
Runo
HTTP server digest algorithm: md5
HTTP server access class: 0
HTTP server base path:
HTTP File Upload status: Disabled
HTTP server upload path:
HTTP server help root:
Maximum number of concurrent server connections allowed: 5
Server idle time-out: 180 seconds
Server life time-out: 180 seconds
Maximum number of requests allowed on a connection: 1
Server linger time : 60 seconds
HTTP server active session modules: ALL
HTTP secure server capability: Present
HTTP secure server status: Disabled
HTTP secure server port: 443
HTTP secure server ciphersuite: 3des-ede-cbc-sha des-cbc-sha rc4-128-md5
rc4-128-sha aes-128-cbc-sha aes-256-cbc-sha dhe-aes-128-cbc-sha
dhe-aes-256-cbc-sha
HTTP secure server client authentication: Disabled
HTTP secure server trustpoint:
HTTP secure server active session modules: ALL

HTTP server application session modules:
  Session module Name Handle Status Secure-status Description
HOME_PAGE 2 Active Active IOS Homepage Server
HTTP_IPS 1 Active Active HTTP based IOS File Server
http_ezsetup 3 Active Active HTTP EZSETUP Server
QDM 4 Active Active QOS Device Manager Server
QDM_SA 5 Active Active QOS Device Manager Signed Applet Server
NBAR2 6 Active Active NBAR2 HTTP(S) Server
WEB_EXEC 7 Active Active HTTP based IOS EXEC Server
IXI 8 Active Active IOS XML Infra Application Server
IDCONF 9 Active Active IDCONF HTTP(S) Server
IPS_SDEE 10 Active Active IOS IPS SDEE Server
EzVPN-Web-Intercept 11 Active Active EzVPN Web Intercept URL Handler
tti-petitioner 12 Active Active TTI Petitioner

HTTP server current connections:
local-ipaddress:port remote-ipaddress:port in-bytes out-bytes

HTTP server statistics:
Accepted connections total: 4

HTTP server history:
local-ipaddress:port remote-ipaddress:port in-bytes out-bytes end-time
172.16.160.142:80 172.16.180.120:41548 333 192 06:00:32 12/20
172.16.160.142:80 172.16.180.120:41550 376 192 06:00:57 12/20
172.16.160.142:80 172.16.180.120:41552 376 192 06:01:08 12/20
172.16.160.142:80 172.16.180.120:41554 376 192 06:01:14 12/20

HTTP server help path:

Runo#write
Building configuration...
[OK]
*Dec 20 06:01:43.730: %GRUB-5-CONFIG_WRITING: GRUB configuration is being updated on disk. Please wait...
Runo#
*Dec 20 06:01:44.927: %GRUB-5-CONFIG_WRITTEN: GRUB configuration was written to disk successfully.
Runo#show ip interface bried
```

Ilustración 4 Comprobación de servicio http

Al desde la máquina virtual accedes por medio del navegador a la dirección del router, este refleja las peticiones realizadas por medio del servicio HTTP, y se podrán consultar por medio del comando show ip http server all, y se desplegará la información de la anterior imagen.

TFTP

Para la implementación del servicio TFTP en la topología, se debe montar y configurar el servidor en una de las máquinas virtuales que estamos utilizando. El procedimiento para dicha actividad se explica a continuación.

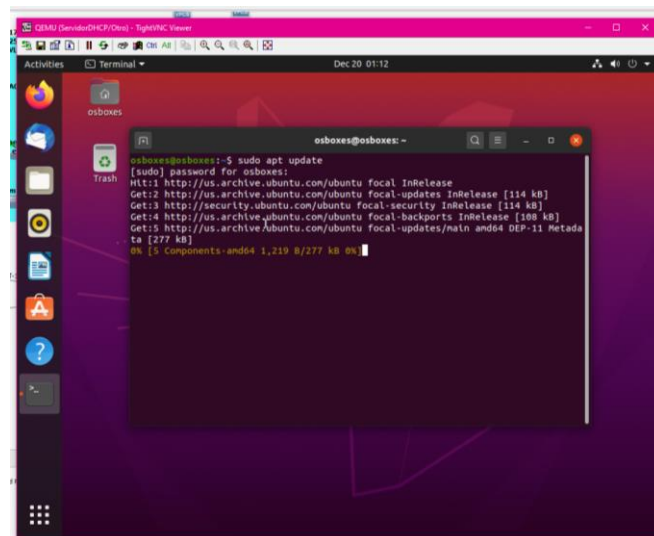


Ilustración 5 Actualización de cache para tftp

Primero, hay que actualizar la caché del repositorio de paquetes APT de la máquina virtual elegida. En este caso es con la máquina virtual de ubuntu. En la ilustración 5 se muestra que para dicha acción hay que ejecutar el comando **sudo apt update**.

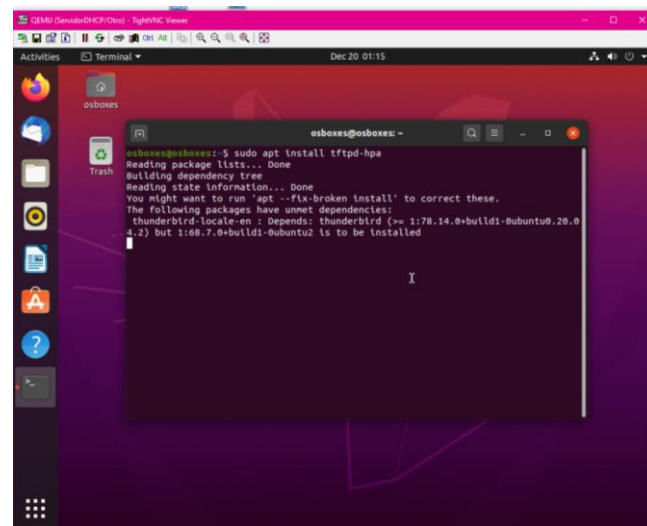
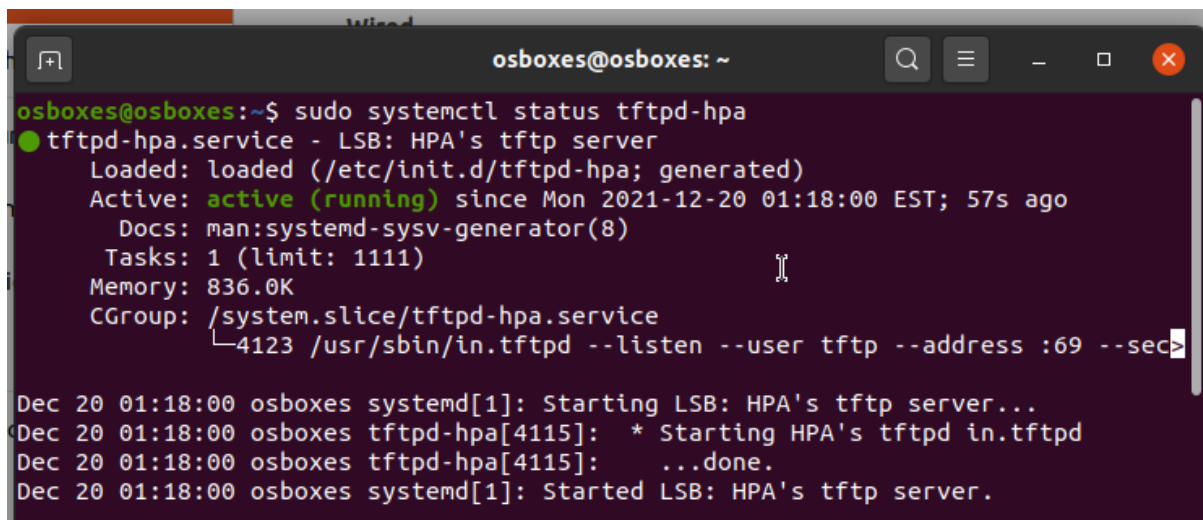


Ilustración 6 Descarga de tftp

Una vez actualizado, podemos descargar e instalar en la máquina el paquete **tftpd-hpa** con el comando **sudo apt install tftpd-hpa**, tal y como se ve en la ilustración 6.



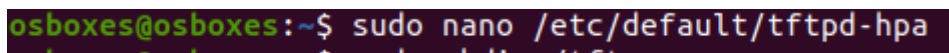
```
osboxes@osboxes:~$ sudo systemctl status tftpd-hpa
● tftpd-hpa.service - LSB: HPA's tftp server
   Loaded: loaded (/etc/init.d/tftpd-hpa; generated)
   Active: active (running) since Mon 2021-12-20 01:18:00 EST; 57s ago
     Docs: man:systemd-sysv-generator(8)
    Tasks: 1 (limit: 1111)
   Memory: 836.0K
    CGroup: /system.slice/tftpd-hpa.service
            └─4123 /usr/sbin/in.tftpd --listen --user tftp --address :69 --sec

Dec 20 01:18:00 osboxes systemd[1]: Starting LSB: HPA's tftp server...
Dec 20 01:18:00 osboxes tftpd-hpa[4115]: * Starting HPA's tftp in.tftpd
Dec 20 01:18:00 osboxes tftpd-hpa[4115]:   ...done.
Dec 20 01:18:00 osboxes systemd[1]: Started LSB: HPA's tftp server.
```

Ilustración 7 Verificar status tftp

Ya que el paquete ha sido instalado es momento de revisar si el servicio de **tftpd-hpa** está corriendo con el comando **sudo systemctl status tftpd-hpa**.

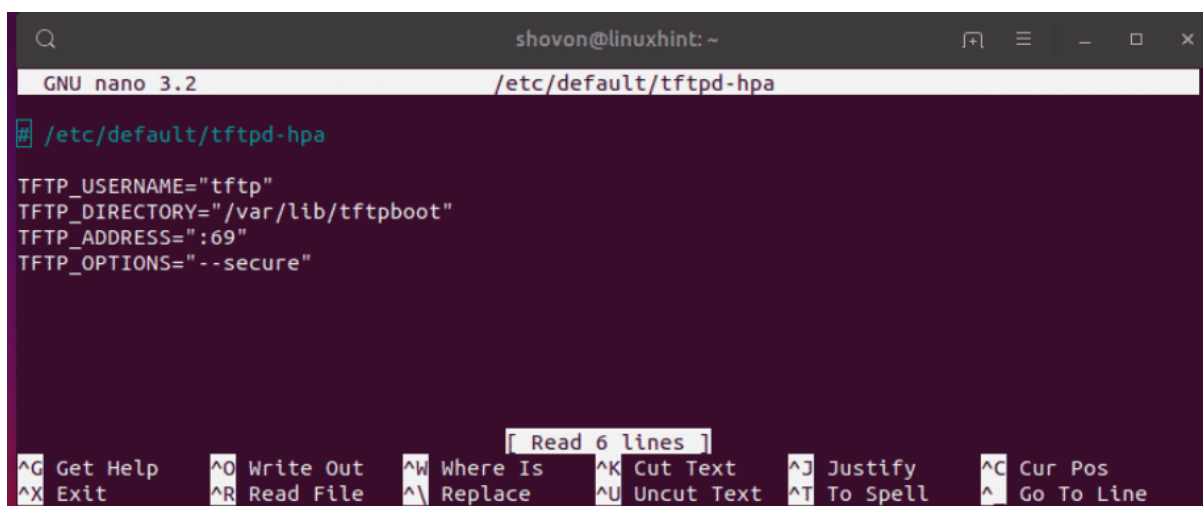
En la ilustración 7 podemos ver cómo se está ejecutando el servicio **tftpd-hpa** y está funcionando correctamente.



```
osboxes@osboxes:~$ sudo nano /etc/default/tftpd-hpa
```

Ilustración 8 Modificación del archivo tftp 1

El siguiente paso es configurarlo, para esto debemos modificar el archivo **tftpd-hpa**. Esto lo hacemos con el comando **sudo nano /etc/default/tftpd-hpa** como se muestra en la ilustración 8.



```
shovon@linuxhint: ~
GNU nano 3.2 /etc/default/tftpd-hpa

/etc/default/tftpd-hpa

TFTP_USERNAME="tftp"
TFTP_DIRECTORY="/var/lib/tftpboot"
TFTP_ADDRESS=":69"
TFTP_OPTIONS="--secure"

Read 6 lines
^G Get Help  ^O Write Out  ^W Where Is   ^K Cut Text   ^J Justify    ^C Cur Pos
^X Exit      ^R Read File  ^\ Replace    ^U Uncut Text ^T To Spell   ^_ Go To Line
```

Ilustración 9 Modificación de archivo tftp 2

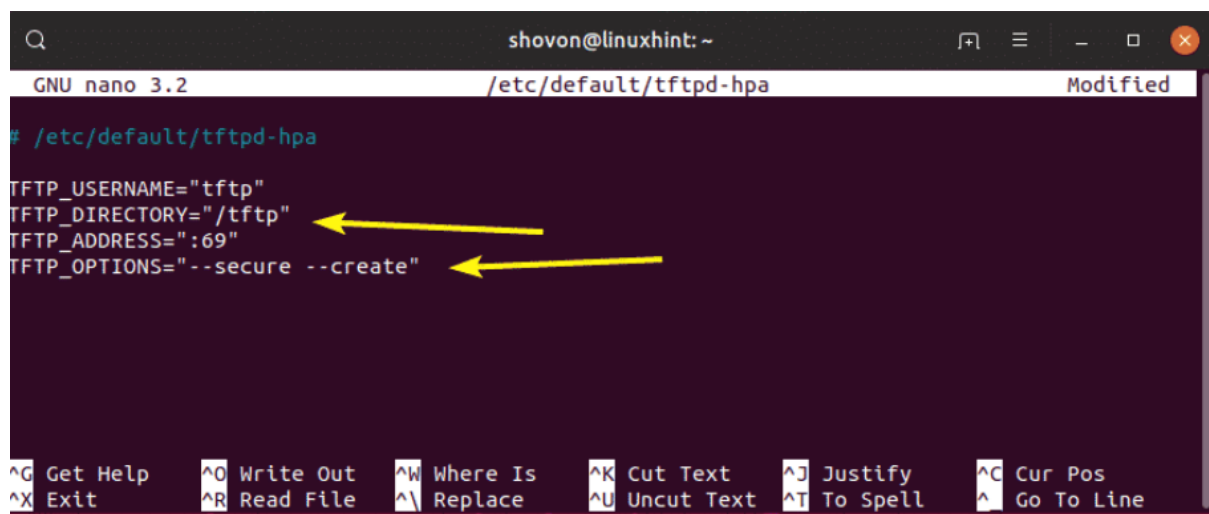
En la ilustración 9 se puede ver el archivo de configuración, debe abrirse para editarlo. Ésta es la configuración predeterminada del servidor TFTP.

Aquí, TFTP_USERNAME se establece en tftp . Significa que el servidor TFTP se ejecutará como el usuario tftp .

TFTP_DIRECTORY está configurado en / var / lib / tftpboot . Significa que / var / lib / tftpboot es el directorio de este servidor al que podrá acceder a través de TFTP.

TFTP_ADDRESS está configurado en : 69 . Significa que TFTP se ejecutará en el puerto 69 .

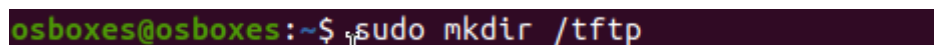
TFTP_OPTIONS está configurado en --secure . Esta variable establece las opciones de TFTP. Hay muchas opciones que puede utilizar para configurar cómo se comportará el servidor TFTP. La opción --secure significa cambiar el directorio TFTP a lo que está configurado en la variable TFTP_DIRECTORY cuando se conecta al servidor TFTP automáticamente. Esta es una característica de seguridad. Si no había configurado la opción --secure , entonces tendría que conectarse al servidor TFTP y configurar el directorio manualmente. Lo cual es muy complicado e inseguro.



```
shovon@linuxhint: ~  
GNU nano 3.2 /etc/default/tftpd-hpa Modified  
# /etc/default/tftpd-hpa  
TFTP_USERNAME="tftp"  
TFTP_DIRECTORY="/tftp"  
TFTP_ADDRESS=":69"  
TFTP_OPTIONS="--secure --create"
```

Ilustración 10 Modificación de archivo tftp 3

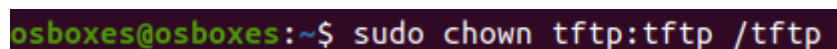
Después de modificar el archivo quedaría cómo se muestra en la ilustración __.



```
osboxes@osboxes:~$ sudo mkdir /tftp
```

Ilustración 11 Creación de directorio tftp

Ahora, debe crear un nuevo directorio /tftp . Para hacer eso, ejecute el siguiente comando: **sudo mkdir / tftp** (ilustración 11).



```
osboxes@osboxes:~$ sudo chown tftp:tftp /tftp
```

Ilustración 12 Cambio de propietario del directorio tftp

Es **importante** cambiar el propietario y el grupo del directorio creado a **tftp** como se puede ver en la ilustración 12.

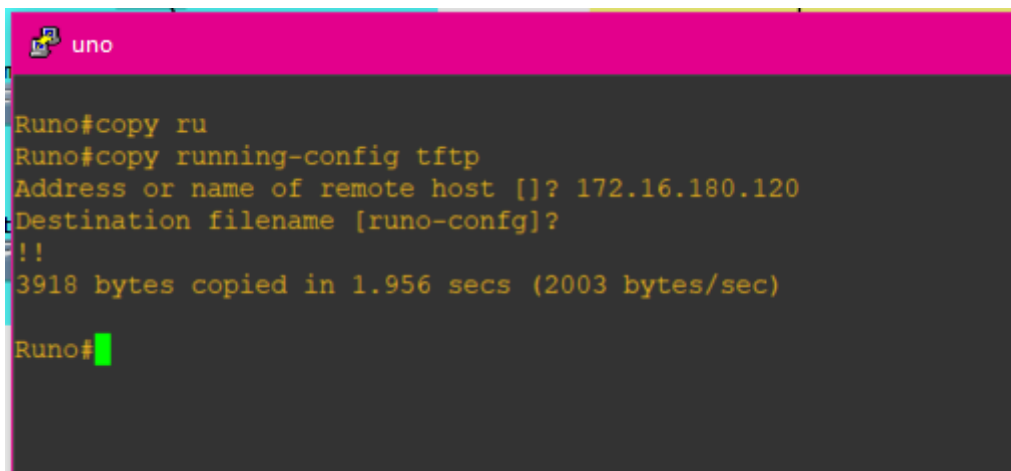
```

osboxes@osboxes:~$ sudo systemctl restart tftpd-hpa
osboxes@osboxes:~$ sudo systemctl status tftpd-hpa
● tftpd-hpa.service - LSB: HPA's tftp server
   Loaded: loaded (/etc/init.d/tftpd-hpa; generated)
   Active: active (running) since Mon 2021-12-20 01:24:15 EST; 2s ago
     Docs: man:systemd-sysv-generator(8)
  Process: 5360 ExecStart=/etc/init.d/tftpd-hpa start (code=exited, status=0/
    Tasks: 1 (limit: 1111)
   Memory: 668.0K
    CGroup: /system.slice/tftpd-hpa.service
            └─5378 /usr/sbin/in.tftpd --listen --user tftp --address :69 --sec
Dec 20 01:24:15 osboxes systemd[1]: Starting LSB: HPA's tftp server...
Dec 20 01:24:15 osboxes tftpd-hpa[5360]: * Starting HPA's tftpd in.tftpd
Dec 20 01:24:15 osboxes tftpd-hpa[5360]:   ...done.
Dec 20 01:24:15 osboxes systemd[1]: Started LSB: HPA's tftp server.

```

Ilustración 13 Reinicio y verificación tftp

Por último paso en la máquina virtual tenemos que reiniciar y revisar nuevamente el estatus del servicio para verificar que funcione correctamente y se hayan guardado los cambios (ilustración 13).



```

Runo#copy ru
Runo#copy running-config tftp
Address or name of remote host []? 172.16.180.120
Destination filename [runo-config]?
!!
3918 bytes copied in 1.956 secs (2003 bytes/sec)
Runo#

```

Ilustración 14 Comprobación de servicio tftp

Para terminar, podemos verificar si se está dando el servicio a través de un router, introduciendo los comandos que se muestran en la ilustración 14. Ahí podemos ver que el envío del archivo es exitoso y el servicio TFTP funciona correctamente.

DHCP

Una vez instalada la máquina virtual con Ubuntu 20.04, se aplica el comando **sudo apt-get update** para obtener todos los paquetes de Ubuntu actualizados. Después instalamos los paquetes necesarios para poder alojar un servidor DHCP con el siguiente comando **sudo apt-get install isc-dhcp-server**, con ello, se nos crean diferentes archivos entre los cuales se destaca el archivo llamado **dhcpd.conf** que se encuentra ubicado en el directorio **/etc/dhcp/**.

Podemos abrirlo con un editor de textos de su preferencia, en este caso se usó Gedit escribiendo el siguiente comando en una terminal, **sudo gedit /etc/dhcp/dhcpd.conf**. Una

vez abierto el archivo **dhcp.conf** se encuentran varias líneas comentadas que son una especie de guía. En ese archivo escribimos lo siguiente:

```
default-lease-time 600;
max-lease-time 7200;

subnet 172.16.180.0 netmask 255.255.252.0 {
option routers 172.16.180.1;
option broadcast-address 172.16.183.255;
}

#AZUL
#VLAN10
subnet 172.16.112.0 netmask 255.255.254.0 {
range 172.16.112.2 172.16.113.254;
option routers 172.16.112.1;
option broadcast-address 172.16.113.255;
}
#VLAN20
subnet 172.16.16.0 netmask 255.255.252.0 {
range 172.16.16.2 172.16.19.254;
option routers 172.16.16.1;
option broadcast-address 172.16.19.255;
}
#VLAN30
subnet 172.16.32.0 netmask 255.255.252.0 {
range 172.16.32.2 172.16.35.254;
option routers 172.16.32.1;
option broadcast-address 172.16.35.255;
}

#AMARILLO
#VLAN10
subnet 172.16.128.0 netmask 255.255.254.0 {
range 172.16.128.2 172.16.129.254;
```

Ilustración 15 Modificación de archivo para DHCP 1


```

option routers 172.16.128.1;
option broadcast-address 172.16.129.255;
}
#VLAN20
subnet 172.16.48.0 netmask 255.255.252.0 {
range 172.16.48.2 172.16.51.254;
option routers 172.16.48.1;
option broadcast-address 172.16.51.255;
}
#VLAN30
subnet 172.16.64.0 netmask 255.255.252.0 {
range 172.16.64.2 172.16.67.254;
option routers 172.16.64.1;
option broadcast-address 172.16.67.255;
}

#ROJO?
#VLAN10
subnet 172.16.144.0 netmask 255.255.254.0 {
range 172.16.144.2 172.16.145.254;
option routers 172.16.144.1;
option broadcast-address 172.16.145.255;
}
#VLAN20
subnet 172.16.80.0 netmask 255.255.252.0 {
range 172.16.80.2 172.16.83.254;
option routers 172.16.80.1;
option broadcast-address 172.16.83.255;
}
#VLAN20
subnet 172.16.96.0 netmask 255.255.252.0 {
range 172.16.96.2 172.16.99.254;
option routers 172.16.96.1;
}

```

Ilustración 16 Modificación de archivo para DHCP 2

```

option routers 172.16.96.1;
option broadcast-address 172.16.99.255;
}

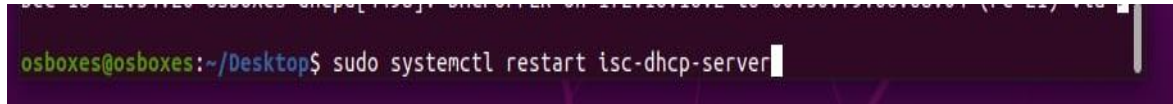
```

Ilustración 17 Modificación de archivo para DHCP 3

El archivo **dhcp.conf** contiene todas las subredes que necesiten una asignación IP dinámica que se encuentran dentro de la topología. Dentro de cada subred especificada, se le asigna un rango disponible de IP de acuerdo con la máscara de esa subred, excluyendo la dirección de broadcast, el puerto de enlace de esa subred y la IP de identificación. Después se le indica el puerto de enlace de esa misma subred y por último la dirección de broadcast.

Se guarda el archivo, y después vamos a la configuración del cable de red que se encuentra dentro de la aplicación “Configuración” de Ubuntu y asignamos un IP estático tomando como referencia nuestra topología. En este caso se le asignó la dirección **172.16.184.120/22**.

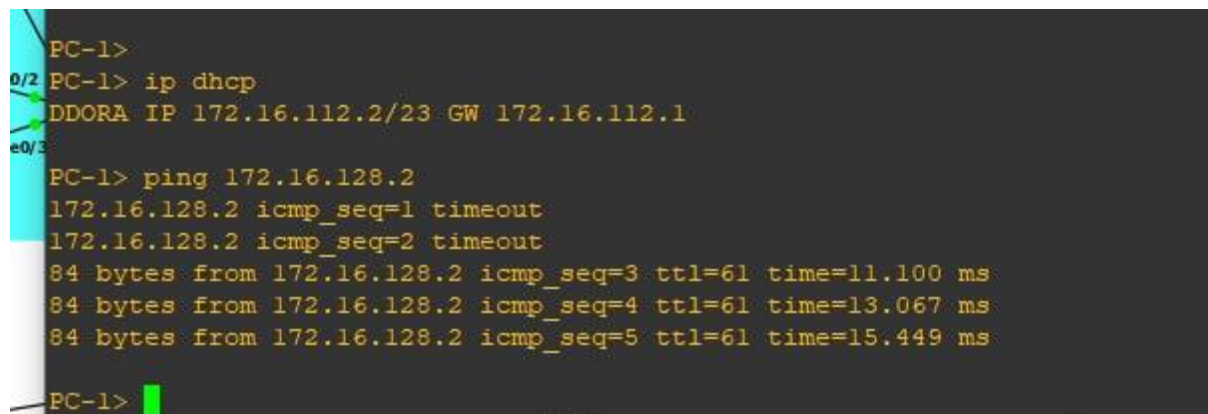
Después se inicia el servidor con el comando **sudo systemctl restart isc-dhcp-server**, para revisar si inició sin errores se verifica con el comando **sudo systemctl status isc-dhcp-server**. Si no hay errores, nuestro servidor DHCP está funcionando.



```
osboxes@osboxes:~/Desktop$ sudo systemctl restart isc-dhcp-server
```

Ilustración 18 Reinicio del servidor DHCP

Para hacer uso de nuestro servidor basta con que la máquina tenga habilitado la opción de DHCP o en una máquina virtual sencilla escribir el comando **ip dhcp**.



```
PC-1>
0/2 PC-1> ip dhcp
DDORA IP 172.16.112.2/23 GW 172.16.112.1
e0/3
PC-1> ping 172.16.128.2
172.16.128.2 icmp_seq=1 timeout
172.16.128.2 icmp_seq=2 timeout
84 bytes from 172.16.128.2 icmp_seq=3 ttl=61 time=11.100 ms
84 bytes from 172.16.128.2 icmp_seq=4 ttl=61 time=13.067 ms
84 bytes from 172.16.128.2 icmp_seq=5 ttl=61 time=15.449 ms
PC-1>
```

Ilustración 19 Prueba DHCP

Conclusión

Como equipo, podemos decir que realizar la planeación, construcción y configuración de una topología (basada en un contexto ideado por nosotros) que ofrece varios servicios, utilizando un protocolo y direccionamiento predeterminado y que además necesita de la implementación de ciertos mecanismos de seguridad como NAT, VLANs y ACLs fue un trabajo desafiante, interesante y gratificante para nuestro conocimiento.

Pensamos que llevar a la práctica lo aprendido durante el curso a un panorama mayor y a través de un emulador como GNS3 fue un reto para la mayoría de nosotros, ya que algunos habíamos trabajado un poco él y para otros era su primera vez interactuando en este sistema. Pero, a través de investigaciones y los tutoriales brindados por la profesora pudimos salir adelante con el proyecto.

Otro problema que se presentó en constantes ocasiones con el proyecto fueron los problemas de almacenamiento de las máquinas virtuales y errores del propio emulador, pero al final pudimos hacer lo mejor posible para afrontar dichas situaciones.

Por otro lado, creemos ampliamente que realizar este proyecto nos ayudó mucho para reforzar los conocimientos adquiridos a lo largo del semestre y mejor aún, conocer cosas nuevas para la implementación de una red y sus servicios.

En conjunto concluimos que el proyecto se finalizó exitosamente cubriendo los requerimientos propuestos al inicio de este.

Fuentes de consulta

[1] https://www.cisco.com/c/es_mx/support/docs/dial-access/asynchronous-connections/5466-comm-server.html

[2] <https://www.cisco.com/c/en/us/td/docs/ios-xml/ios/https/command/nm-https-cr-book/nm-https-cr-cl-sh.html#wp2028963790>

[3] https://linuxhint.com/install_tftp_server_ubuntu/

[4] <https://martinsblog.dk/linux-how-do-i-enable-snmp-on-ubuntu/>

[5] <https://computingforgeeks.com/install-and-configure-cacti-on-ubuntu/>