

Escuela Colombiana de Ingeniería Julio Garavito

Redes de Cómputo

Laboratorio N° 6

Protocolos capa de aplicación, Plataforma base y capa de red

Integrantes:

Laura Sofia Gil Chaves

Camilo Castaño Quintanilla

Docentes:

Ing. Claudia Patricia Santiago Cely

Ing. John Pachón

20 de abril de 2024

1. Introducción

2. Desarrollo del Tema

2.1 Marco Teórico

2.2 Uso de aplicaciones

2.2.1 Simulación con Routers

2.2.2 Instalación de software base

2.2.2.1 Instalación de servicio web

2.2.2.2 Configuración de servicio de hosting

3. Conclusiones

4. Bibliografía

1. Introducción

En el desarrollo de servicios web y hosting, se integran estaciones de usuario tanto alámbricas como inalámbricas junto con servidores (físicos y virtualizados). Estos elementos están interconectados mediante switches de capa 2 y 3, equipos inalámbricos y routers que facilitan la conexión a Internet. Además, es común utilizar infraestructuras en la nube para la provisión dinámica de recursos según las necesidades de la organización. Dentro de los servidores dedicados al servicio web y hosting, se alojan diversos servicios como páginas web, DNS, correo electrónico, bases de datos, almacenamiento y aplicaciones, entre otros, proporcionando así un entorno completo y seguro para la gestión y distribución de contenidos en línea.

2. Desarrollo del Tema

2.1 Marco Teórico

En la actualidad, los servicios web y el hosting son componentes fundamentales para el funcionamiento de numerosas aplicaciones y sitios web en línea. Estos servicios permiten a las organizaciones y usuarios individuales compartir información, acceder a recursos y ofrecer experiencias interactivas en la web. Ahora exploraremos diferentes tecnologías utilizadas para implementar servicios web, centrándonos en Apache en OpenBSD, Nginx en Slackware y la configuración en Windows Server, así como también abordaremos la configuración de servicios de hosting.

En primer lugar, definiremos lo qué es un servicio web, donde es una aplicación o recurso disponible en internet que facilita la interacción entre distintos sistemas informáticos o dispositivos. Funciona mediante estándares de comunicación como HTTP y utiliza formatos de intercambio de datos como XML o JSON. Esto permite que aplicaciones desarrolladas en diferentes plataformas puedan comunicarse de manera eficiente y transparente, gracias a interfaces bien definidas y protocolos estandarizados.

Apache es uno de los servidores web más populares y ampliamente utilizados en el mundo. En el entorno de OpenBSD, un sistema operativo conocido por su enfoque en la seguridad y el rendimiento, Apache ofrece una solución robusta y confiable para alojar aplicaciones web y sitios estáticos o dinámicos. La configuración de Apache en OpenBSD se basa en la integración de sus módulos, la gestión de la seguridad mediante reglas de firewall y la optimización del rendimiento para garantizar una experiencia de usuario fluida y segura.

Por otra parte, tenemos Nginx que es otro servidor web popular conocido por su eficiencia y escalabilidad en el manejo de cargas pesadas. En el contexto de Slackware, una distribución de Linux reconocida por su simplicidad y estabilidad, Nginx ofrece una alternativa potente para el alojamiento de aplicaciones web. La configuración de Nginx en Slackware se enfoca en la optimización de recursos, la distribución de carga y la implementación de técnicas avanzadas como

el almacenamiento en caché y la compresión de datos para mejorar el rendimiento y la velocidad de respuesta del servidor.

En el ecosistema de Windows Server, se pueden implementar servicios web mediante la configuración de Server Manager que se centra en herramientas de desarrollo y administración, la gestión de certificados de seguridad, el control de acceso y la optimización de recursos para garantizar la disponibilidad y el rendimiento de los servicios web alojados en la plataforma.

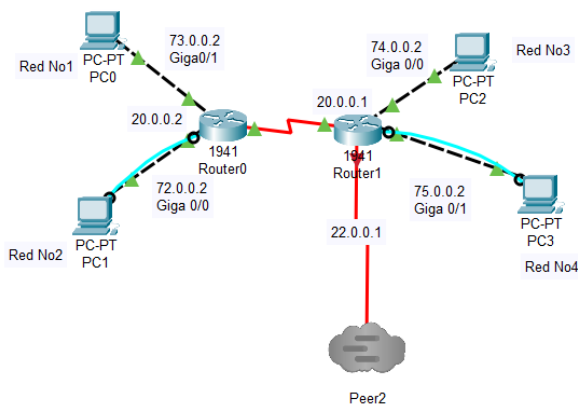
Por último, tenemos la configuración de servicios de hosting que abarca aspectos como la asignación de recursos, la gestión de dominios y subdominios, la configuración de bases de datos, la implementación de medidas de seguridad (firewalls) y la disponibilidad de los servicios alojados.

2.2 Uso y Aplicaciones

2.2.1 Simulación con Routers

Packet Tracer Sofia

Creamos la red establecida en el laboratorio y hacemos las configuraciones necesarias para cada uno de los elementos siguiendo la guía de laboratorio.



Prendemos todo lo de routers y pcs

```
Router>enable
Router#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#interface GigabitEthernet0/0
Router(config-if)#no shutdown

Router(config-if)#
%LINK-5-CHANGED: Interface GigabitEthernet0/0, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/0, changed state to
up

Router(config-if)#exit
Router(config)#interface GigabitEthernet0/1
Router(config-if)#no shutdown

Router(config-if)#
%LINK-5-CHANGED: Interface GigabitEthernet0/1, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/1, changed state to
up
```

Probamos la redes para diferente LAN

```
C:\>ping 72.0.0.2

Pinging 72.0.0.2 with 32 bytes of data:

Reply from 72.0.0.2: bytes=32 time=23ms TTL=255
Reply from 72.0.0.2: bytes=32 time<1ms TTL=255
Reply from 72.0.0.2: bytes=32 time<1ms TTL=255

Ping statistics for 72.0.0.2:
    Packets: Sent = 3, Received = 3, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 23ms, Average = 7ms

Control-C
^C
C:\>tracert 73.0.0.2

Tracing route to 73.0.0.2 over a maximum of 30 hops:

  1  0 ms    0 ms    0 ms    73.0.0.2

Trace complete.
```

Para pc0 me conecto a PC1

```
ping 73.0.0.2

Pinging 73.0.0.2 with 32 bytes of data:

Reply from 73.0.0.2: bytes=32 time=3ms TTL=255
Reply from 73.0.0.2: bytes=32 time<1ms TTL=255

Ping statistics for 73.0.0.2:
    Packets: Sent = 2, Received = 2, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 3ms, Average = 1ms

Control-C
^C
C:\>tracert 72.0.0.1

Tracing route to 72.0.0.1 over a maximum of 30 hops:

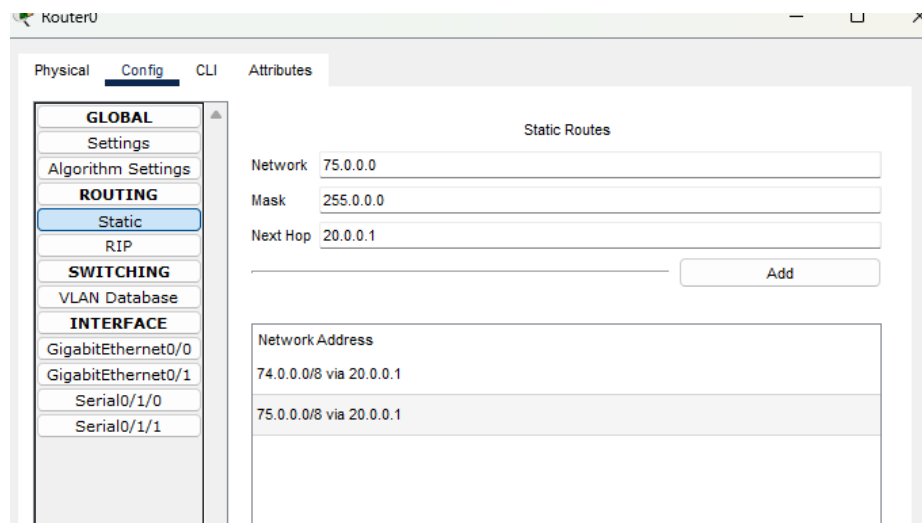
  1  0 ms    0 ms    0 ms    73.0.0.2
  2  0 ms    0 ms    0 ms    72.0.0.1

Trace complete.

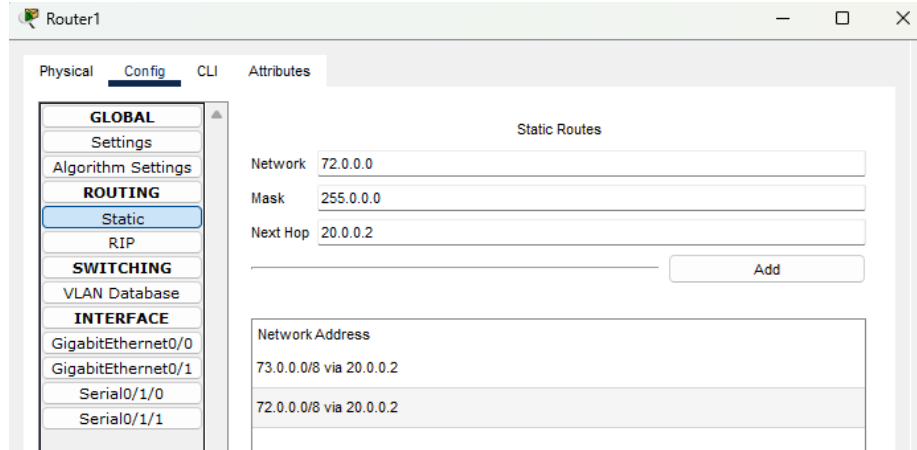
C:\>
```

Ahora configuramos los routers con rutas estática de tal manera que pueda hacer ping entre todos los equipos de la red (Todos los pc0 se pueden ver entre si).

Para Router 0



Para Router 1



Mediante el comando ICMP traceroute, se verifica la comunicación entre los computadores.

```

C:\>tracert 74.0.0.1

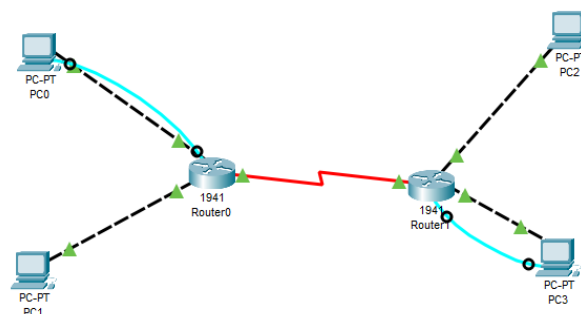
Tracing route to 74.0.0.1 over a maximum of 30 hops:

  0  7 ms    0 ms    0 ms    73.0.0.2
  1  1 ms    4 ms    2 ms    20.0.0.1
  2  *        3 ms    4 ms    74.0.0.1

Trace complete.
  
```

Packet Tracer Camilo

Creamos la red como se nos pide



Desde PC3 configuramos el Router 0 (milo) y guardamos

```
Router>enable
Router#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#hostname milo
milo(config)#banner motd # Uso Exclusivo de Estudiantes de Reco #
milo(config)#line console 0
milo(config-line)#logging synchronous
milo(config-line)#password Clave_E
milo(config-line)#login
milo(config-line)#exit
milo(config)#vty 0 15
^
% Invalid input detected at '^' marker.

milo(config)#vty 0 15
^
% Invalid input detected at '^' marker.

milo(config)#line vty 0 15
milo(config-line)#logging synchronous
milo(config-line)#Clave_T
^
% Invalid input detected at '^' marker.

milo(config-line)#password Clave_T
milo(config-line)#login
milo(config-line)#exit
milo(config)#no ip domain-lookup
milo(config)#interface GigabitEthernet0/0
milo(config-if)#description "descripcion1"
milo(config-if)#exit
milo(config)#enable secret Clave_E
milo(config)#exit
milo#
%SYS-5-CONFIG_I: Configured from console by console

milo#
```

```
^
% Invalid input detected at '^' marker.

milo(config-line)#password Clave_T
milo(config-line)#login
milo(config-line)#exit
milo(config)#no ip domain-lookup
milo(config)#interface GigabitEthernet0/0
milo(config-if)#description "descripcion1"
milo(config-if)#exit
milo(config)#enable secret Clave_E
milo(config)#exit
milo#
%SYS-5-CONFIG_I: Configured from console by console

milo#show running-config
Building configuration...

Current configuration : 982 bytes
!
version 15.1
no service timestamps log datetime msec
no service timestamps debug datetime msec
no service password-encryption
!
hostname milo
!
!
!
enable secret 5 $1$mERr$Fok00o4D/yfqmsnEIPSG7.
!
!
!
!
no ip cef
no ipv6 cef
--More--
```

```
milo#copy running-config startup-config
Destination filename [startup-config]?
Building configuration...
[OK]
```

Repetimos el proceso con Router1


```

Terminal
Router>enable
Router#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#hostname castano
castano(config)#banner motd # mensaje #
castano(config)#line console 0
castano(config-line)#logging synchronous
castano(config-line)#password Clave_E
castano(config-line)#login
castano(config-line)#exit
castano(config)#line vty 0 15
castano(config-line)#logging synchronous
castano(config-line)#password Clave_T
^
% Invalid input detected at '^' marker.

castano(config-line)#password Clave_T
castano(config-line)#login
castano(config-line)#exit
castano(config)#no ip domain-lookup
castano(config)#interface GigabitEthernet0/0
^
% Invalid input detected at '^' marker.

castano(config)#interface GigabitEthernet0/0
castano(config-if)#description "PC-3"
castano(config-if)#exit
castano(config)#enable secret Clave_E
castano(config)#exit
castano#
%SYS-5-CONFIG_I: Configured from console by console

castano#

castano(config-line)#exit
castano(config)#no ip domain-lookup
castano(config)#interface GigabitEthernet0/0
^
% Invalid input detected at '^' marker.

castano(config)#interface GigabitEthernet0/0
castano(config-if)#description "PC-3"
castano(config-if)#exit
castano(config)#enable secret Clave_E
castano(config)#exit
castano#
%SYS-5-CONFIG_I: Configured from console by console

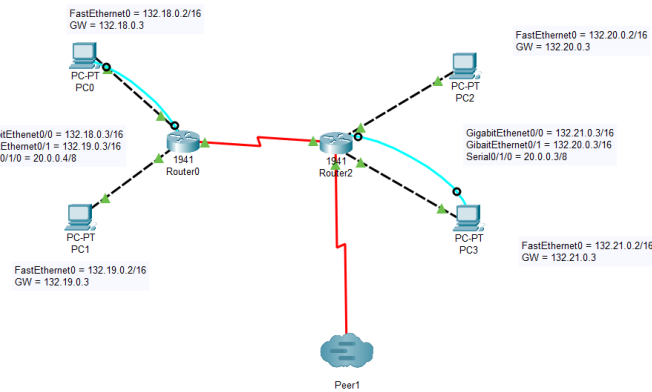
castano#show running-config
Building configuration...

Current configuration : 968 bytes
!
version 15.1
no service timestamps log datetime msec
no service timestamps debug datetime msec
no service password-encryption
!
hostname castano
!
!
!
enable secret 5 $1$mERr?Fok00o4D/yfqmsnEIPSG7.
!
!
!
!
no ip cef
no ipv6 cef
!
castano#

castano#copy running-config startup-config
Destination filename [startup-config]?
Building configuration...
[OK]
castano#

```

Asignamos las direcciones IP a tanto a los computadores, como a los routers de la red, teniendo en cuenta las direcciones dadas para estudiante1, con las siguientes direcciones.



Al momento de probar la conectividad a los equipos de la misma LAN (el mismo pc y el Router o entre los routers) esta se establece perfectamente. Sin embargo, al tratar de hacerlo a otras redes, no se puede establecer la comunicación.

```

PC0
Physical Config Desktop Programming Attributes
Command Prompt
C:\>
C:\>
C:\>
C:\>
C:\>
C:\>ping 132.18.0.2

Pinging 132.18.0.2 with 32 bytes of data:

Reply from 132.18.0.2: bytes=32 time=8ms TTL=128
Reply from 132.18.0.2: bytes=32 time=11ms TTL=128
Reply from 132.18.0.2: bytes=32 time=9ms TTL=128
Reply from 132.18.0.2: bytes=32 time=8ms TTL=128

Ping statistics for 132.18.0.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 8ms, Maximum = 11ms, Average = 9ms

C:\>ping 132.18.0.3

Pinging 132.18.0.3 with 32 bytes of data:

Reply from 132.18.0.3: bytes=32 time<1ms TTL=255
Reply from 132.18.0.3: bytes=32 time<1ms TTL=255
Reply from 132.18.0.3: bytes=32 time<1ms TTL=255
Reply from 132.18.0.3: bytes=32 time<1ms TTL=255

Ping statistics for 132.18.0.3:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\>ping 132.19.0.2

Pinging 132.19.0.2 with 32 bytes of data:

Request timed out.
Request timed out.

Ping statistics for 132.19.0.2:
    Packets: Sent = 3, Received = 0, Lost = 3 (100% loss),

Control-C
  
```

```
C:\>ping 132.19.0.2

Pinging 132.19.0.2 with 32 bytes of data:

Request timed out.
Request timed out.

Ping statistics for 132.19.0.2:
    Packets: Sent = 3, Received = 0, Lost = 3 (100% loss),

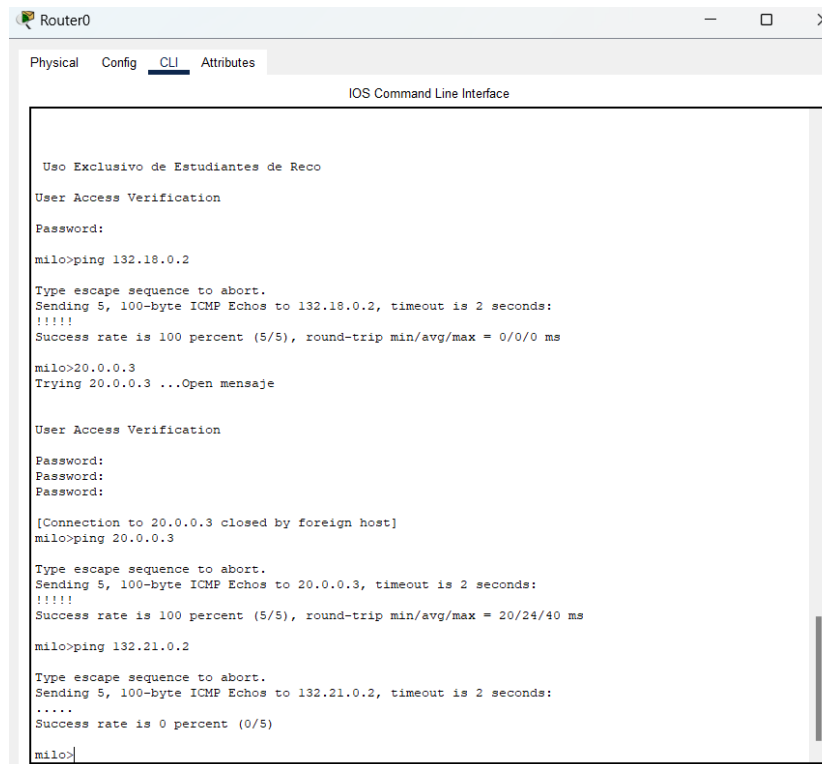
Control-C
^C
C:\>ping 132.19.0.3

Pinging 132.19.0.3 with 32 bytes of data:

Request timed out.

Ping statistics for 132.19.0.3:
    Packets: Sent = 2, Received = 0, Lost = 2 (100% loss),

Control-C
^C
C:\>
```



Configuramos los routers con rutas estaticas

```

milo>enable
Password:
milo#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
milo(config)#ip route 132.20.0.0 255.255.0.0 20.0.0.3
milo(config)#ip route 132.21.0.0 255.255.0.0 20.0.0.3
milo(config)#wr mem
^
% Invalid input detected at '^' marker.

milo(config)#exit
milo#
%SYS-5-CONFIG_I: Configured from console by console

milo#wr mem
Building configuration...
[OK]
milo#show ip route static
S    132.20.0.0/16 [1/0] via 20.0.0.3
S    132.21.0.0/16 [1/0] via 20.0.0.3
milo#

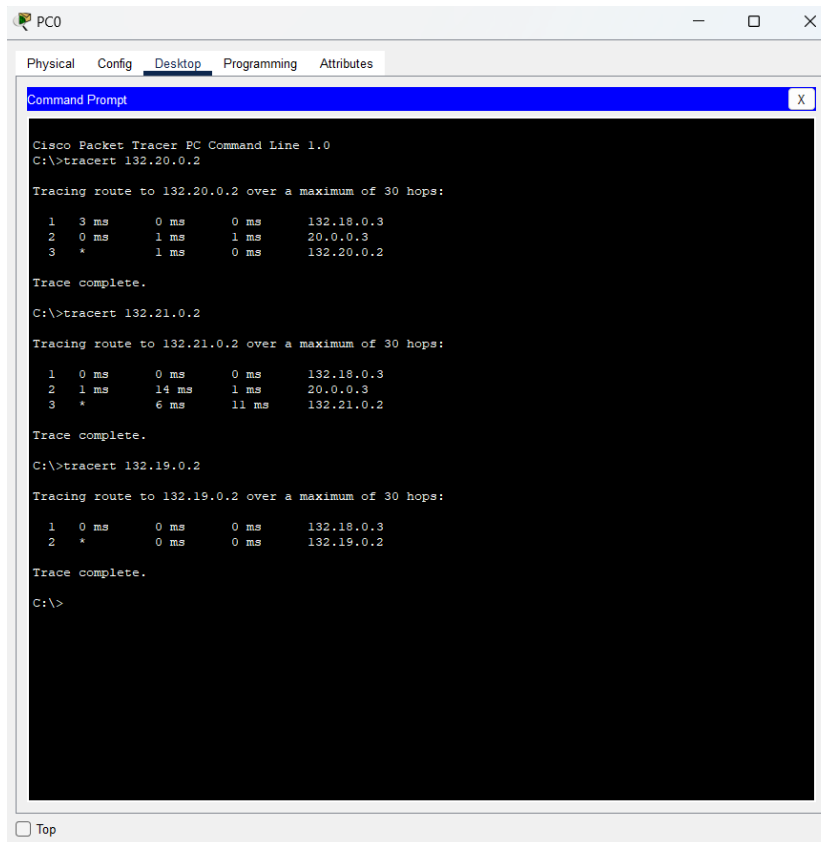
```

```

castano#show ip route static
S    132.18.0.0/16 [1/0] via 20.0.0.4
S    132.19.0.0/16 [1/0] via 20.0.0.4

```

Usamos el comando tracert para verificar la comunicacion entre los dispositivos



```

PC0
Physical Config Desktop Programming Attributes
Command Prompt
Cisco Packet Tracer PC Command Line 1.0
C:\>tracert 132.20.0.2

Tracing route to 132.20.0.2 over a maximum of 30 hops:

  1  3 ms    0 ms    0 ms    132.18.0.3
  2  0 ms    1 ms    1 ms    20.0.0.3
  3  *        1 ms    0 ms    132.20.0.2

Trace complete.

C:\>tracert 132.21.0.2

Tracing route to 132.21.0.2 over a maximum of 30 hops:

  1  0 ms    0 ms    0 ms    132.18.0.3
  2  1 ms   14 ms    1 ms    20.0.0.3
  3  *        6 ms   11 ms    132.21.0.2

Trace complete.

C:\>tracert 132.19.0.2

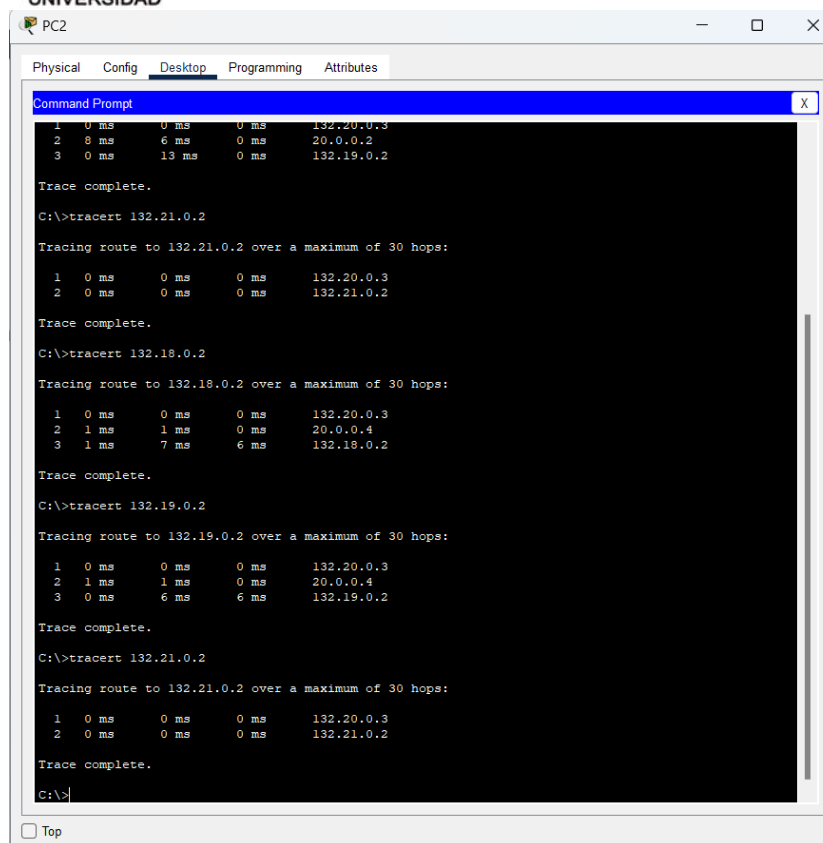
Tracing route to 132.19.0.2 over a maximum of 30 hops:

  1  0 ms    0 ms    0 ms    132.18.0.3
  2  *        0 ms    0 ms    132.19.0.2

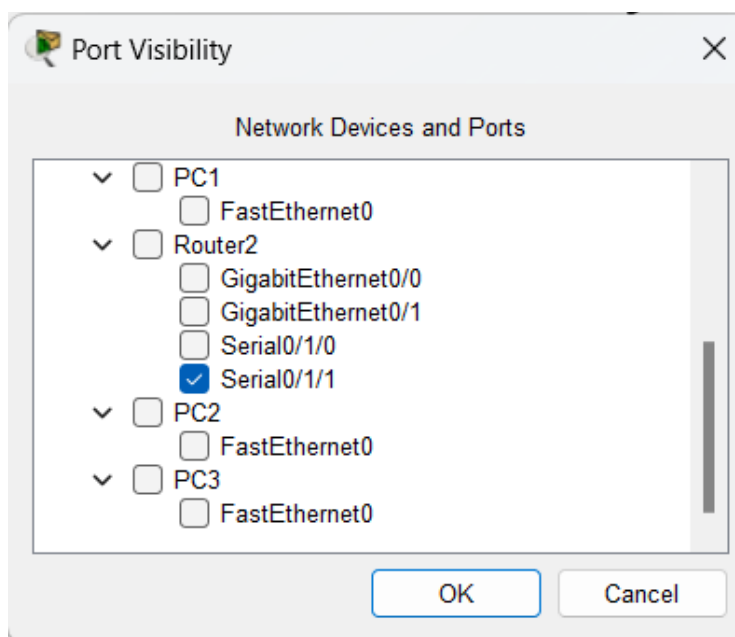
Trace complete.

C:\>

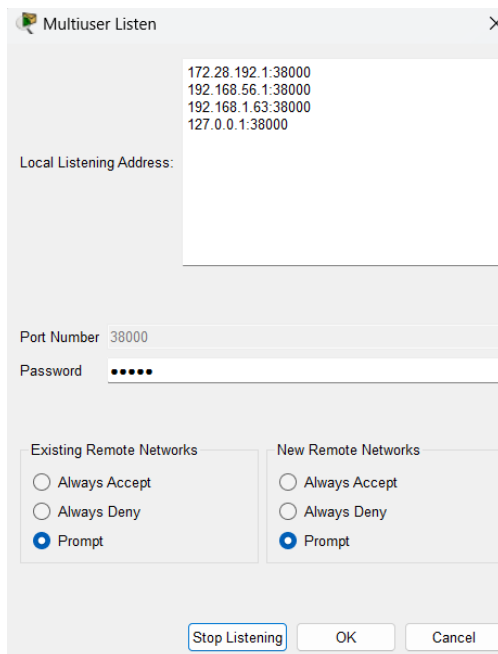
```



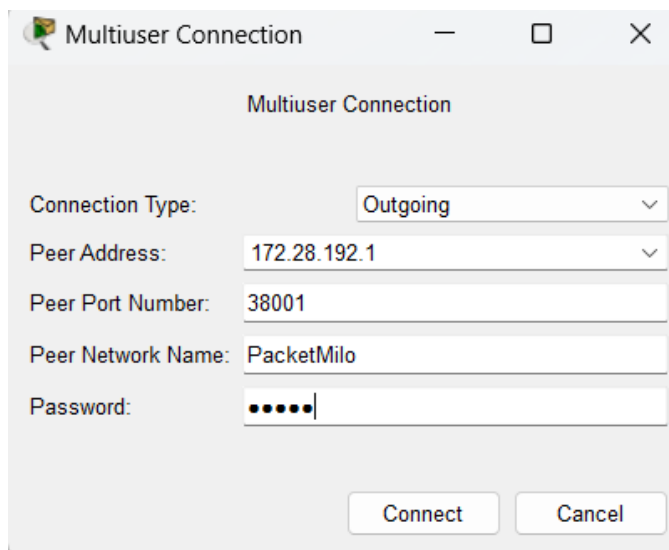
Para montar la conexión multiusuario, seleccionamos que los otros usuarios se pueden conectar al router dos desde Serial0/1/1



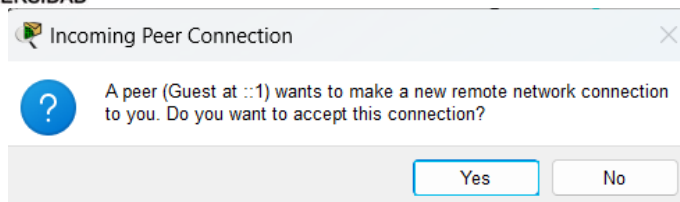
Previamente, establecimos las configuraciones de multiusuario, estableciendo como contraseña CISCO



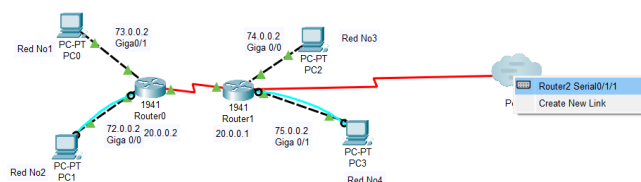
Desde la red de Sofía establecemos la conexión a la red de Camilo



Aceptamos la solicitud de conexión

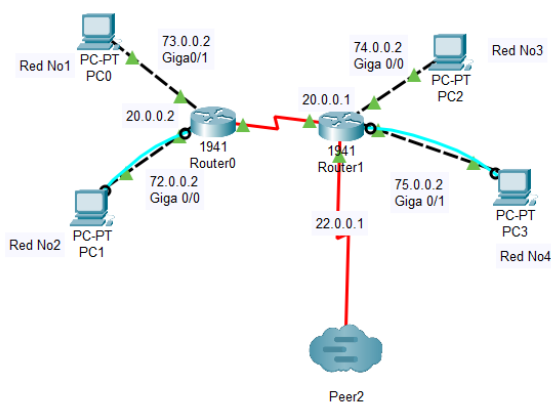


Establecemos una conexión serial entre el router 2 de Camilo y el router 1 de Sofía

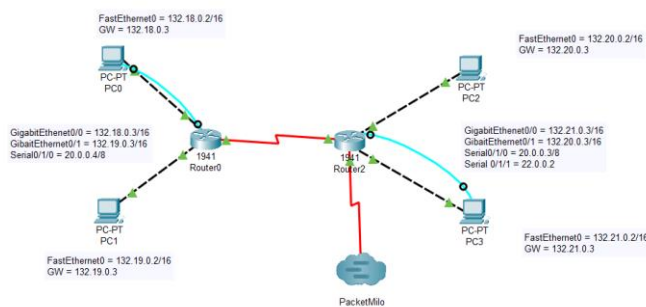


Le asignamos las nuevas IP a los dentro de la red 22.0.0.0/8

Dejando la red de sofia de la siguiente forma



Y la de camilo como



Configuramos las nuevas rutas estaticas

```

<Gil>#show ip rout static
S    72.0.0.0/8 [1/0] via 20.0.0.2
S    73.0.0.0/8 [1/0] via 20.0.0.2
S    132.18.0.0/16 [1/0] via 22.0.0.2
S    132.19.0.0/16 [1/0] via 22.0.0.2
S    132.20.0.0/16 [1/0] via 22.0.0.2
S    132.21.0.0/16 [1/0] via 22.0.0.2
  
```

```

<Sofia>#show ip route static
S    74.0.0.0/8 [1/0] via 20.0.0.1
S    75.0.0.0/8 [1/0] via 20.0.0.1
S    132.18.0.0/16 [1/0] via 20.0.0.1
S    132.19.0.0/16 [1/0] via 20.0.0.1
S    132.20.0.0/16 [1/0] via 20.0.0.1
S    132.21.0.0/16 [1/0] via 20.0.0.1
  
```

```

castano#show ip route static
S    72.0.0.0/8 [1/0] via 22.0.0.1
S    73.0.0.0/8 [1/0] via 22.0.0.1
S    74.0.0.0/16 is subnetted, 1 subnets
S    74.0.0.0 [1/0] via 22.0.0.1
S    75.0.0.0/16 is subnetted, 1 subnets
S    75.0.0.0 [1/0] via 22.0.0.1
S    132.18.0.0/16 [1/0] via 20.0.0.4
S    132.19.0.0/16 [1/0] via 20.0.0.4
  
```

```

milo#show ip route static
S    72.0.0.0/8 [1/0] via 20.0.0.3
S    73.0.0.0/8 [1/0] via 20.0.0.3
S    74.0.0.0/8 [1/0] via 20.0.0.3
S    75.0.0.0/8 [1/0] via 20.0.0.3
S    132.20.0.0/16 [1/0] via 20.0.0.3
S    132.21.0.0/16 [1/0] via 20.0.0.3
  
```

2.2.2.Instalación de Software Base

2.2.2.1 Instalación de servicio web

SERVIO WEB OPENBSD

Instalamos el paquete apache-httpd

```
Last login: Fri Mar 15 16:39:26 on ttyC0
OpenBSD 7.4 (GENERIC) #1336: Tue Oct 10 08:52:22 MDT 2023

Welcome to OpenBSD: The proactively secure Unix-like operating system.

Please use the sendbug(1) utility to report bugs in the system.
Before reporting a bug, please try to reproduce it with the latest
version of the code. With bug reports, please try to ensure that
enough information to reproduce the problem is enclosed, and if a
known fix for it exists, include that as well.

You have new mail.
OPEN# pkg_add apache-httpd
quirks-6.160 signed on 2024-04-02T23:15:13Z
apache-httpd-2.4.58p0:pcr2-10.37p1: ok
apache-httpd-2.4.58p0:ngtcp2-0.19.1: ok
apache-httpd-2.4.58p0:nghttp3-0.15.0: ok
apache-httpd-2.4.58p0:curl-8.6.0: ok
apache-httpd-2.4.58p0:db-4.6.21p7v0: ok
apache-httpd-2.4.58p0:apr-1.7.2: ok
apache-httpd-2.4.58p0:apr-util-1.6.3: ok
apache-httpd-2.4.58p0: ok
The following new rcscripts were installed: /etc/rc.d/apache2
See rcctl(8) for details.
OPEN#
```

Seguidamente lo iniciamos con el comando `/etc/rc.d/apache2 start`

```
OPEN# /etc/rc.d/apache2 start
apache2(ok)
OPEN#
```

Dentro de `/var/www/htdocs` creamos el siguiente html

```
GNU nano 7.2 index.html
<!DOCTYPE html>
<html>
<head>
  <meta charset="UTF-8">
  <title>Hola</title>
</head>
<body>
  <h1 style="margin-bottom: 20px;">El mega proyecto</h1>
  <p style="margin-bottom: 20px;">Noticias de ultima hora, el profesor ha adm
  <img href="italy.jpg" alt="Foto de italia">

</body>
</html>

[ Read 15 lines (Converted from DOS format) ]
G Help      O Write Out  W Where Is   K Cut        T Execute    C Location
X Exit      R Read File  N Replace   U Paste      J Justify    Go To Line
```

Y probamos que sirva el servicio buscando la página por su IP

El mega proyecto

Noticia de última hora, el profesor de RECO a admitido que Camilo y Sofia son el mejor grupo

[Foto de Italo](#)

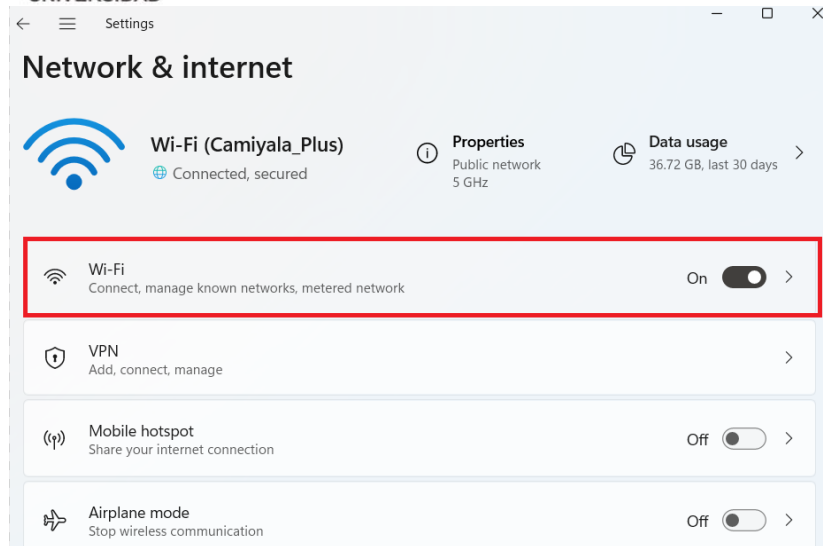
Con rcctl enable, permitimos que cada vez que se prenda la maquina de OPENBSD, se inicie el servicio de apache de forma automática.

```
cron_flags=
dhcpcd_flags=""
dhcpleased_flags=
dhcrelay_flags=NO      # for normal use: "-i interface [server]"
dvmrpd_flags=NO
eigrpd_flags=NO
ftpd_flags=NO          # set to NO if ftpd is running out of inetd
ftpproxy_flags=NO
ftpproxy6_flags=NO
hostapd_flags=NO
hotplugd_flags=NO
httpd_flags=NO
identd_flags=NO
ifstated_flags=NO
iked_flags=NO
inetd_flags=NO

OPENF# /etc/rc.d/apache2 restart
apache2(ok)
apache2(ok)
OPENF# enable apache2
ksh: enable: not found
OPENF# rcctl enable apache2
OPENF#
```

Ahora vamos a configurar en el computador físico el servidor DNS para que en internet podamos acceder al servidor web por nombre en lugar que por su dirección IP.

Lo primero que hacemos es darle click al internet actual



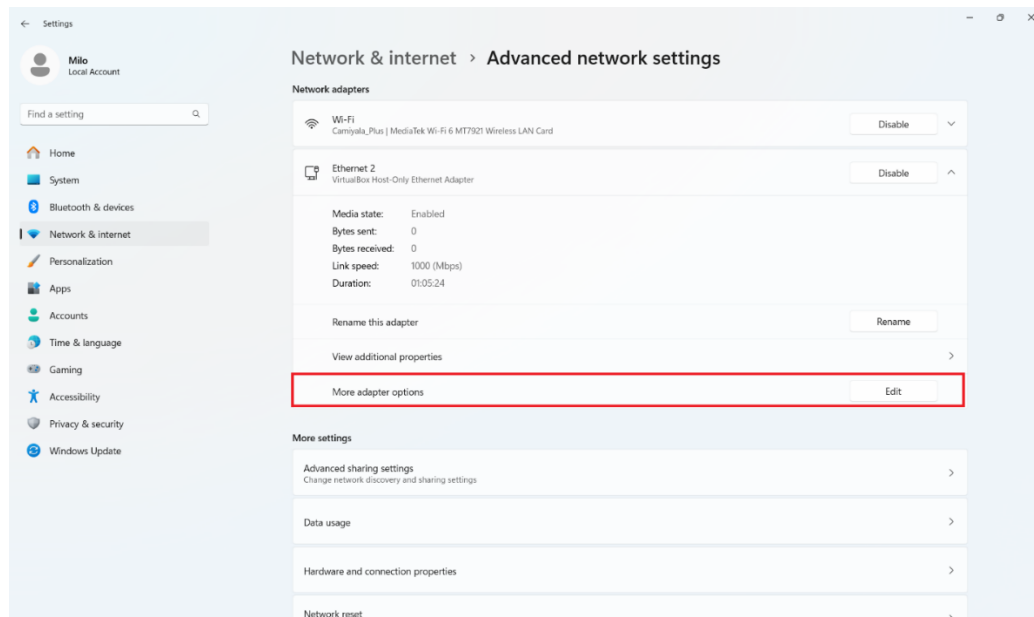
Propiedades

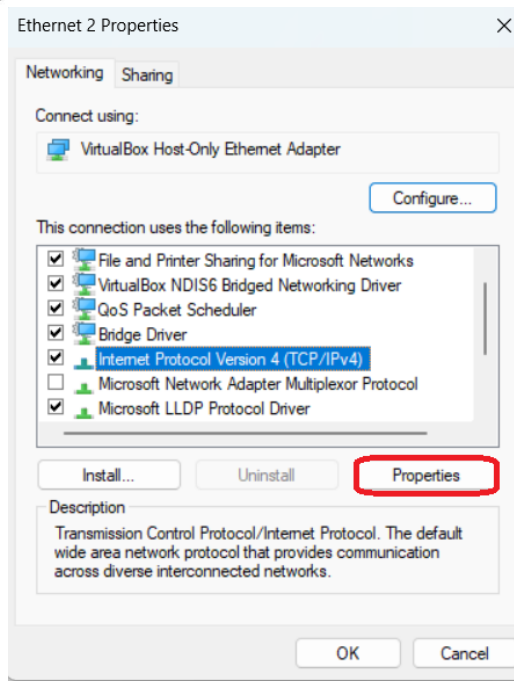


Ajustamos manualmente el servicio DNS

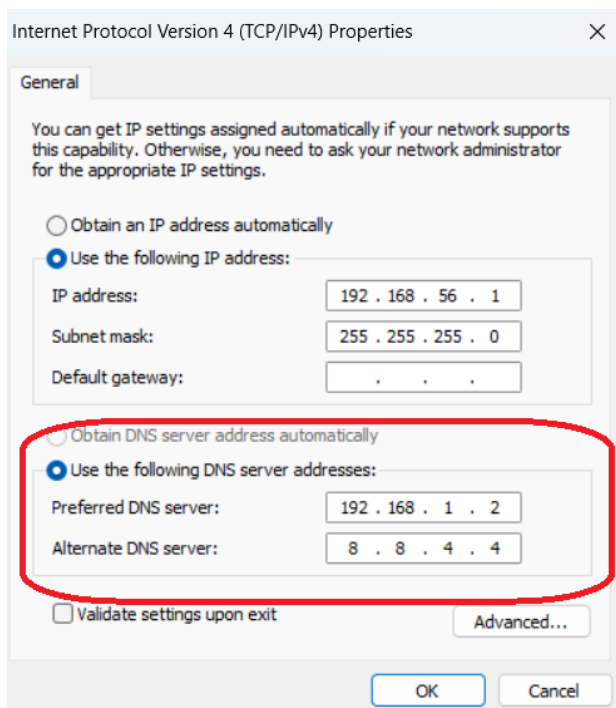


Configuraciones avanzadas

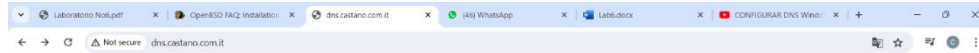




Agregamos el servidor DNS



De esta forma ya podemos buscar el servidor web por su nombre.



El mega proyecto

Noticia de ultima hora, el profesor de RECO a admitido que Camilo y Sofia son el mejor grupo

[Foto de Italia](#)

SERVIDOR WEB WINDOWS

Vamos al Server Manager, a la opción All servers

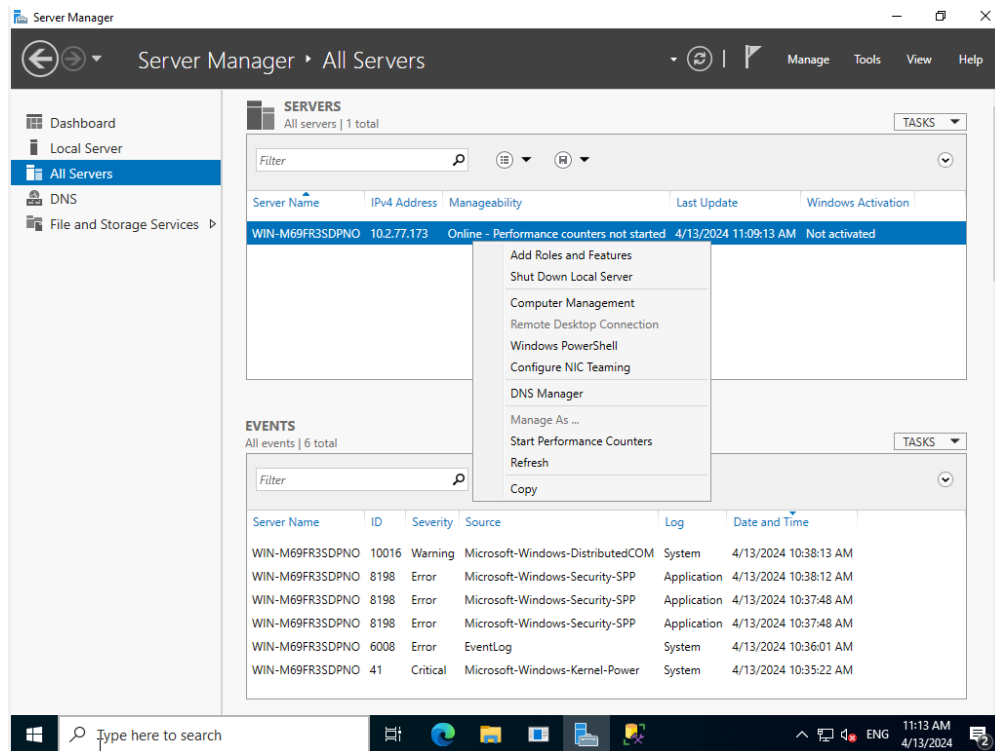
SERVERS
All servers | 1 total

Server Name	IPv4 Address	Manageability	Last Update	Windows Activation
WIN-M69FR3SDPNO	10.2.77.173	Online - Performance counters not started	4/13/2024 11:09:13 AM	Not activated

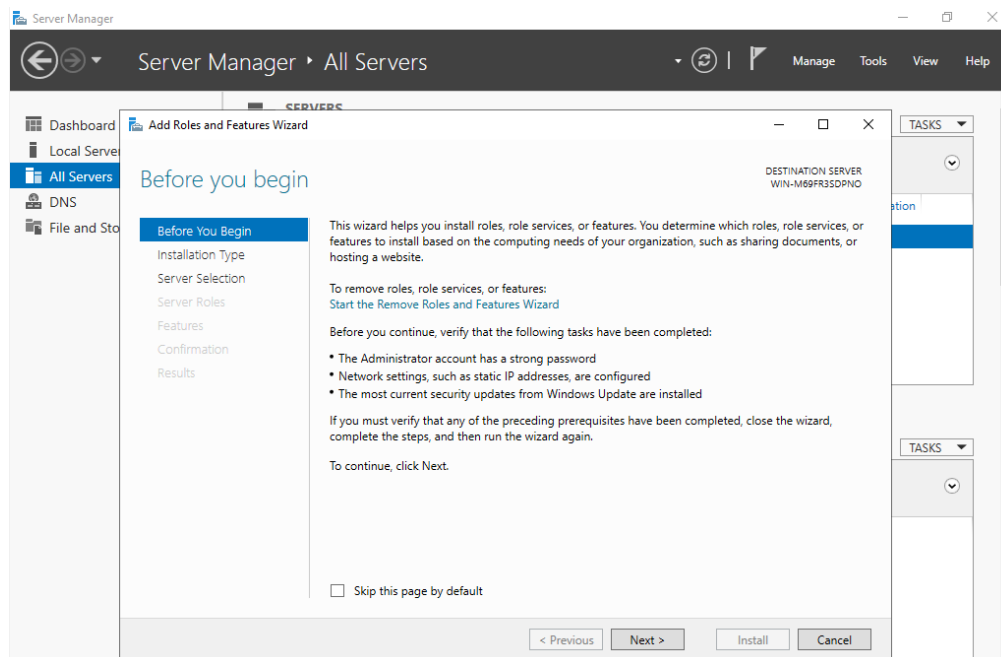
EVENTS
All events | 6 total

Server Name	ID	Severity	Source	Log	Date and Time
WIN-M69FR3SDPNO	10016	Warning	Microsoft-Windows-DistributedCOM	System	4/13/2024 10:38:13 AM
WIN-M69FR3SDPNO	8198	Error	Microsoft-Windows-Security-SPP	Application	4/13/2024 10:38:12 AM
WIN-M69FR3SDPNO	8198	Error	Microsoft-Windows-Security-SPP	Application	4/13/2024 10:37:48 AM
WIN-M69FR3SDPNO	8198	Error	Microsoft-Windows-Security-SPP	Application	4/13/2024 10:37:48 AM
WIN-M69FR3SDPNO	6008	Error	EventLog	System	4/13/2024 10:36:01 AM
WIN-M69FR3SDPNO	41	Critical	Microsoft-Windows-Kernel-Power	System	4/13/2024 10:35:22 AM

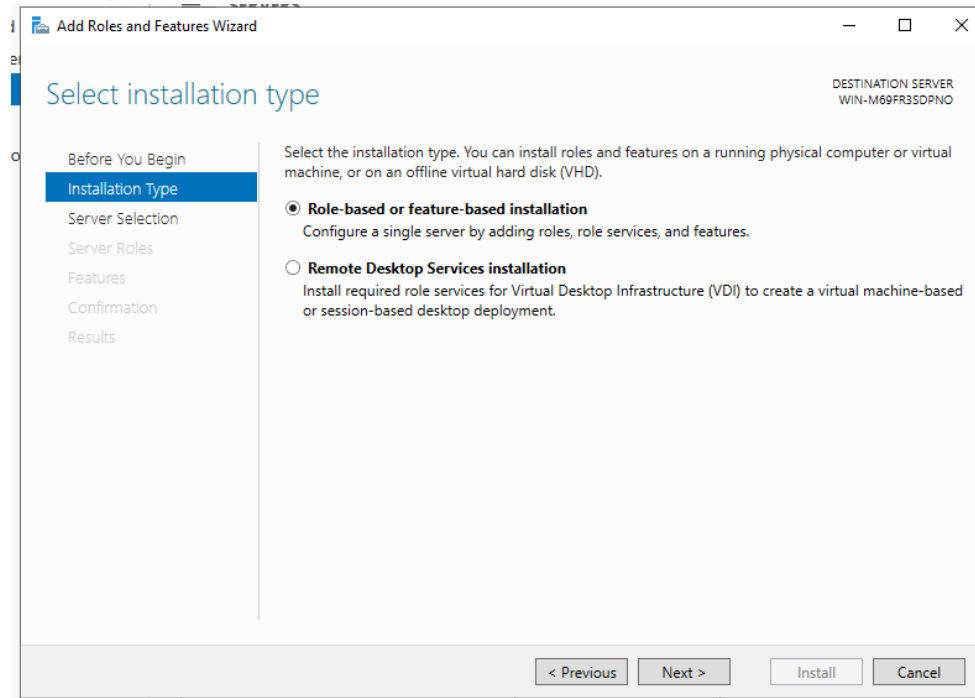
Hacemos Click derecho en el servidor



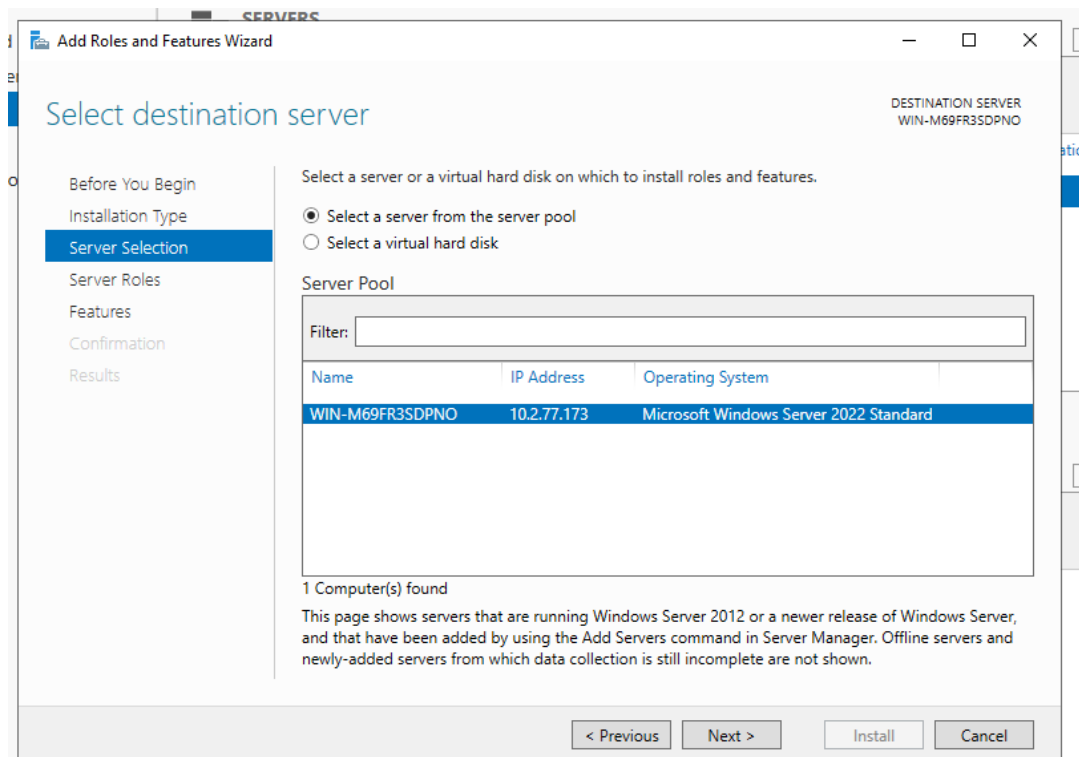
Accedemos a la opción add roles and features



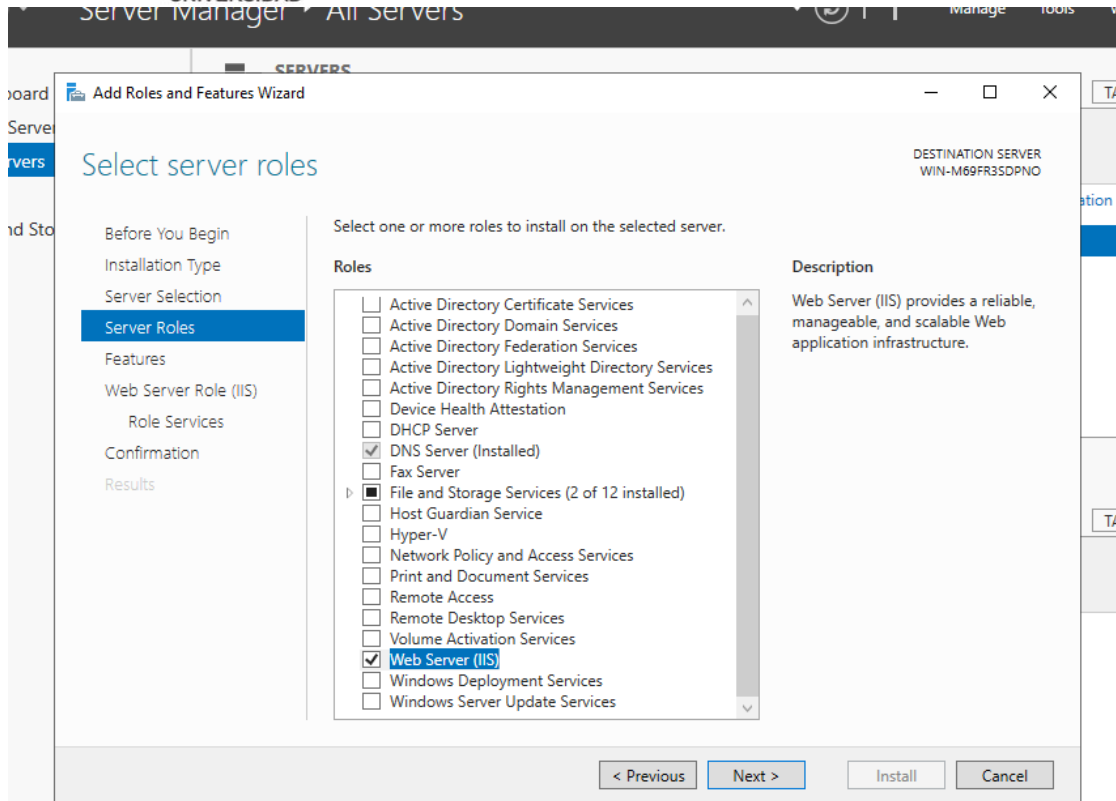
Seleccionamos tipo de instalación (la primera opción) y damos next



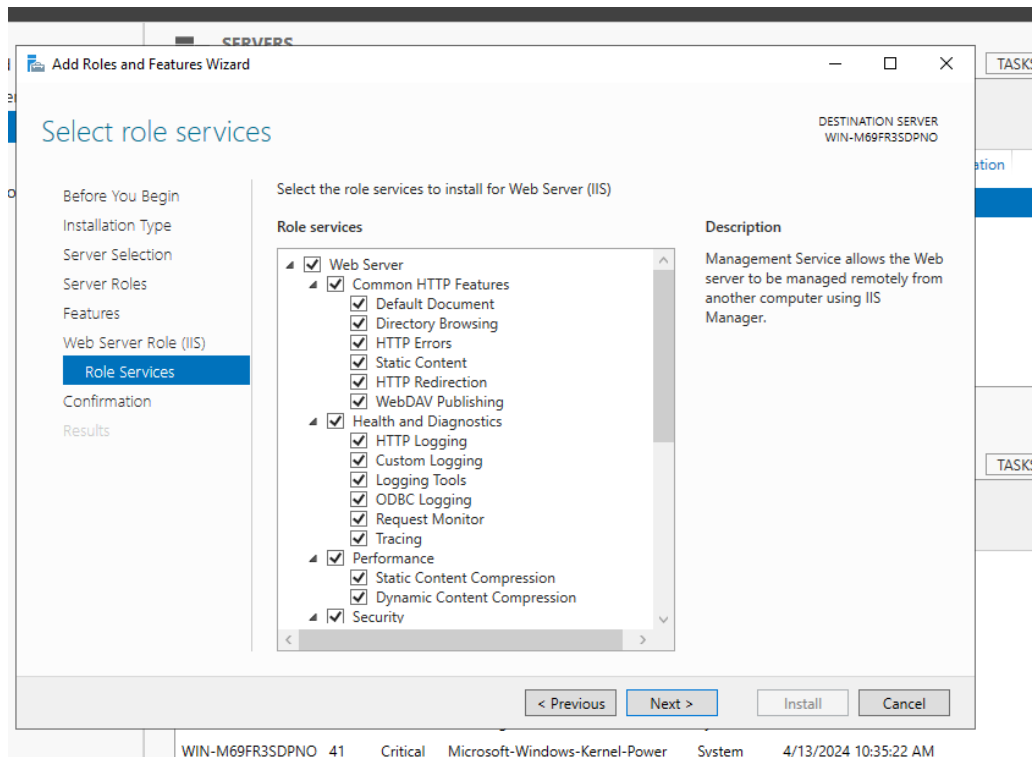
Seleccionamos (la primera opción de server selection) y damos next



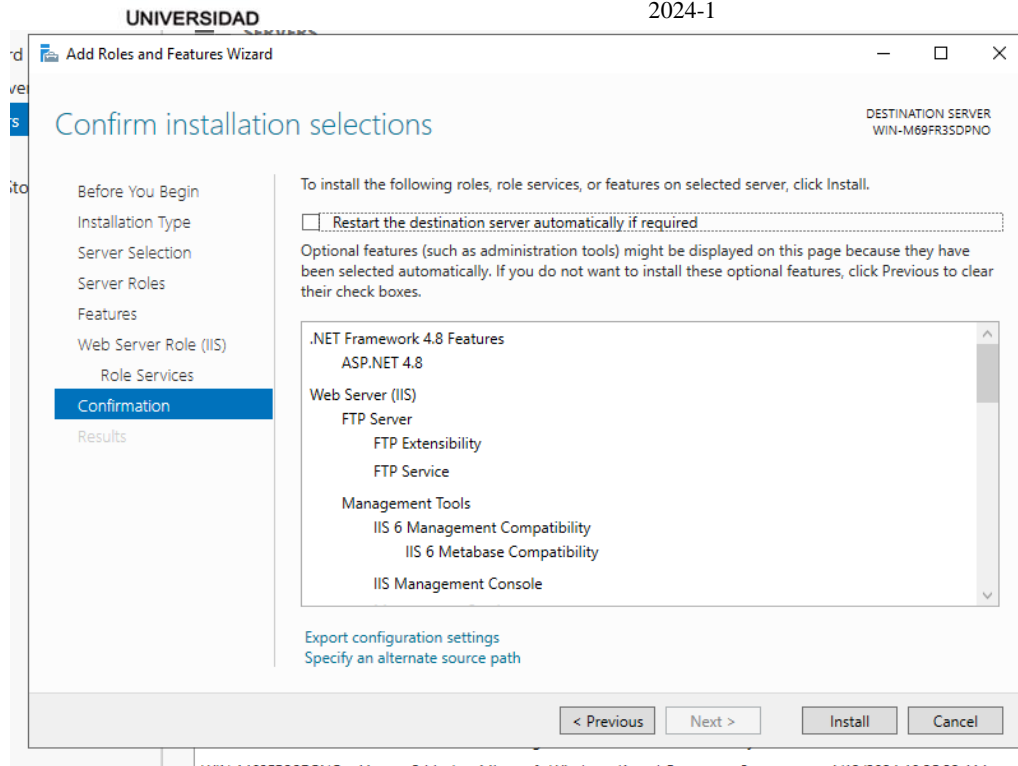
Luego nos dirigimos a server roles y damos en la opción Web Server



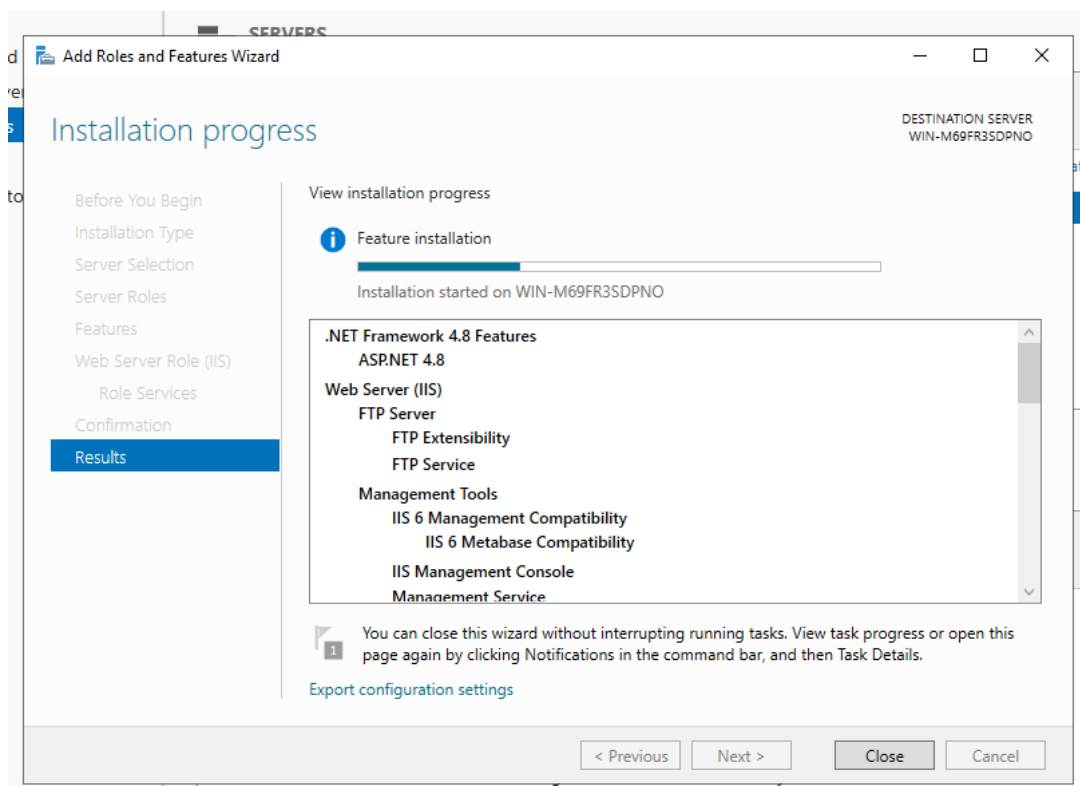
Se seleccionan todas las opciones del web server



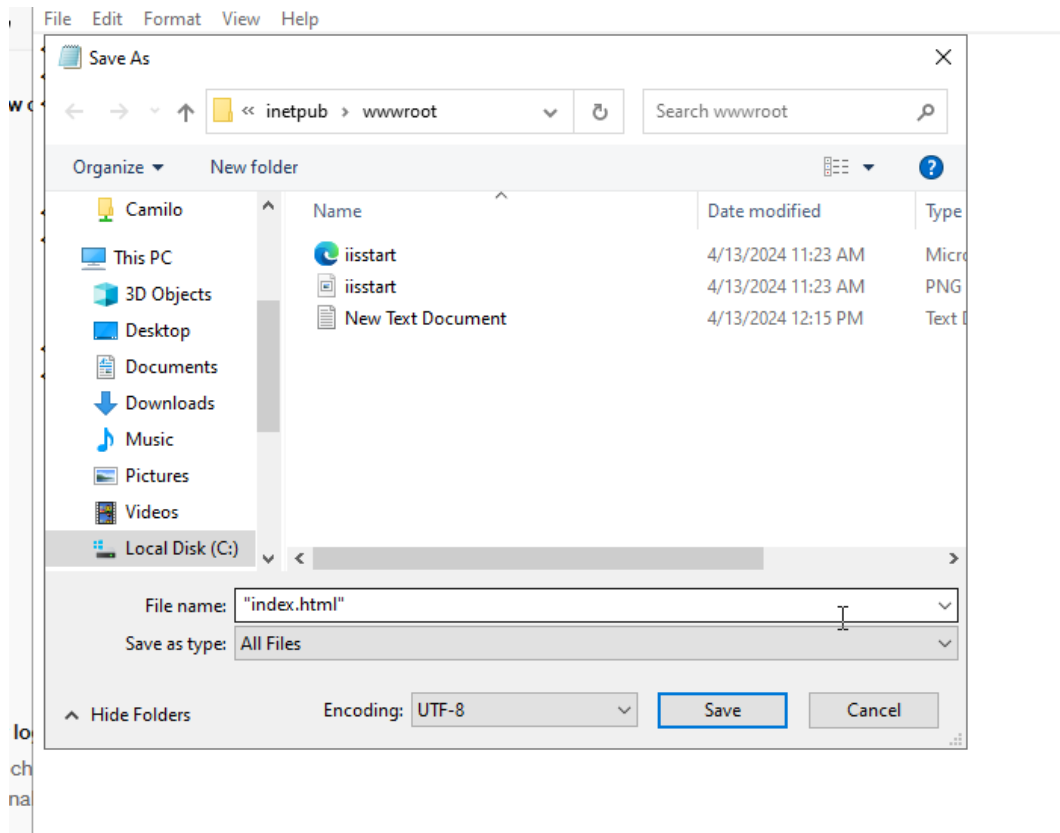
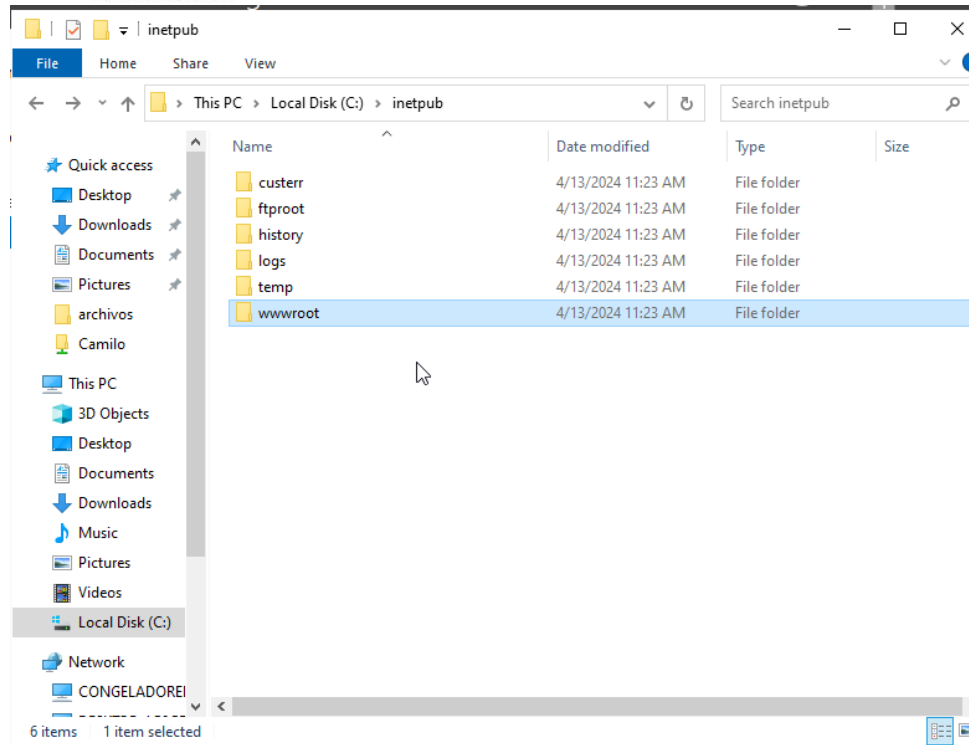
Aceptamos la confirmación de la instalación.

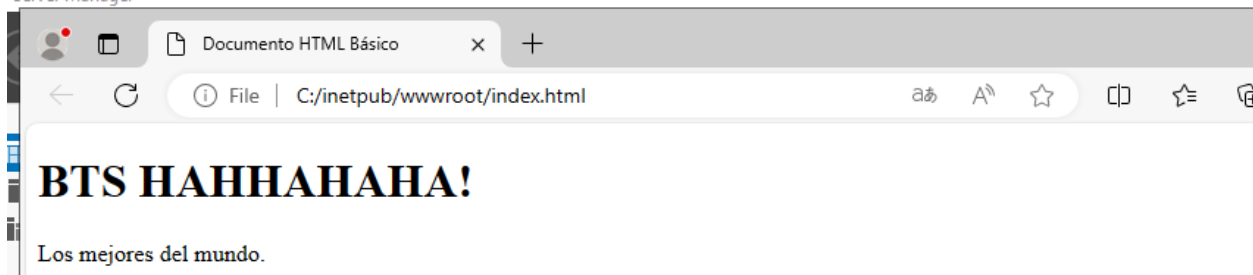


Se instala servicio web



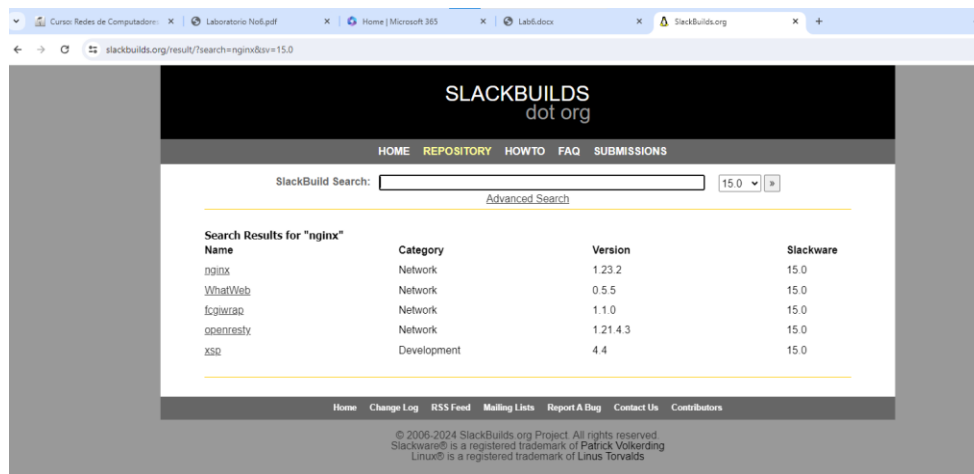
Vamos a disco local c , ingresamos wwwroot y centro de esa carpeta creamos una página web



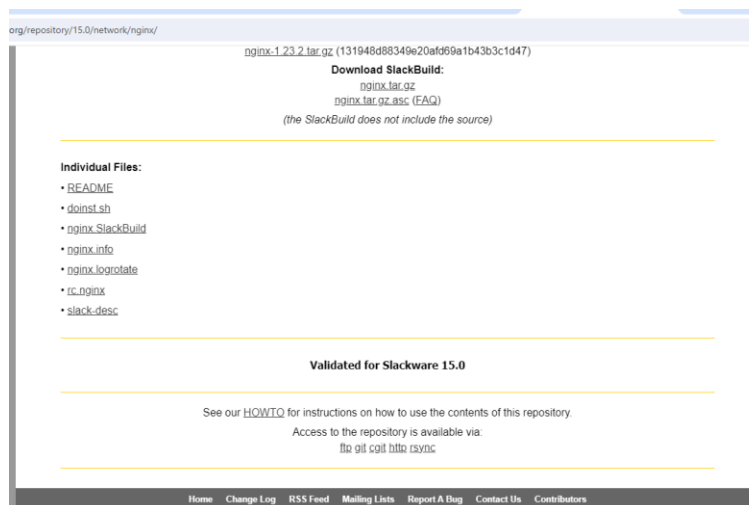


SERVIDOR WEB SLACKWARE

Buscamos en la página nginx y miramos nginx tar gz



Miramos la ruta de instalación del constructor



Descargamos nginx.tar.gz

```

slackwaresql [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Starting OpenSSH SSH daemon: /usr/sbin/sshd
Starting ACPI daemon: /usr/sbin/acpid
Updating MIME database: /usr/bin/update-mime-database /usr/share/mime &
Updating gtk.immodules:
  /usr/bin/update-gtk-immodules &
Updating gdk-pixbuf.loaders:
  /usr/bin/update-gdk-pixbuf-loaders &
Compiling GSettings XML schema files:
  /usr/bin/glib-compile-schemas /usr/share/glib-2.0/schemas &
Starting crond: /usr/sbin/crond -l notice
Starting atd: /usr/sbin/atd -b 15 -l 1
Loading /usr/share/kbd/keymaps/i386/qwerty/la-latin1.map.gz
Starting gpm: /usr/sbin/gpm -m /dev/mouse -t imps2
Starting PostgreSQL
waiting for server to start..... done
server started

Welcome to Linux 5.15.19 x86_64 (tty1)

slack login: root
Password:
Last login: Sat Apr 13 17:05:30 on tty1
Linux 5.15.19.
root@slack:~# wget https://slackbuilds.org/slackbuilds/15.0/network/nginx.tar.gz
--2024-04-15 19:44:11-- https://slackbuilds.org/slackbuilds/15.0/network/nginx.tar.gz
Resolving slackbuilds.org (slackbuilds.org)... 66.85.79.67, 64:ff9b::4255:4f43
Connecting to slackbuilds.org (slackbuilds.org)|66.85.79.67|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 5693 (5.6K) [application/x-gzip]
Saving to: 'nginx.tar.gz'

nginx.tar.gz      100%[=====>]  5.56K  --.-KB/s   in 0s

2024-04-15 19:44:12 (531 MB/s) - 'nginx.tar.gz' saved [5693/5693]

root@slack:~#

```

Ahora descargamos el compilador y el instalador de nginx

```

root@slack:~# wget https://nginx.org/download/nginx-1.23.2.tar.gz
--2024-04-15 19:47:00-- https://nginx.org/download/nginx-1.23.2.tar.gz
Resolving nginx.org (nginx.org)... 3.125.197.172, 52.58.199.22, 64:ff9b::37d:c5ac, ...
Connecting to nginx.org (nginx.org)|3.125.197.172|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 1108243 (1.1M) [application/octet-stream]
Saving to: 'nginx-1.23.2.tar.gz'

nginx-1.23.2.tar.gz  100%[=====>]  1.06M  168KB/s   in 6.5s

2024-04-15 19:47:08 (168 KB/s) - 'nginx-1.23.2.tar.gz' saved [1108243/1108243]

root@slack:~# _

```

Pasamos los archivos anteriormente descargados a un directorio nuevo llamado nginx en el root

```
root@slack:/# ls
bin/  etc/  lib64/  mnt/  nginx.tar.gz  root/  srv/  usr/
boot/ home/ lost+found/ nginx/  opt/  run/  sys/  var/
dev/  lib/  media/  nginx-1.23.2.tar.gz  proc/  sbin/  tmp/
root@slack:/# mv nginx-1.23.2.tar.gz nginx
root@slack:/# ls
bin/  dev/  home/  lib64/  media/  nginx/  opt/  root/  sbin/  sys/  usr/
boot/  etc/  lib/  lost+found/  mnt/  nginx.tar.gz  proc/  run/  srv/  tmp/  var/
root@slack:/# mv nginx.tar.gz nginx
root@slack:/# ls
bin/  dev/  home/  lib64/  media/  nginx/  proc/  run/  srv/  tmp/  var/
boot/  etc/  lib/  lost+found/  mnt/  opt/  root/  sbin/  sys/  usr/
root@slack:/# cd nginx/
root@slack:/nginx# ls
nginx-1.23.2.tar.gz  nginx.tar.gz
root@slack:/nginx# _
```

Descomprimos nginx.tar.gz

```
root@slack:/nginx# ls
nginx-1.23.2.tar.gz  nginx.tar.gz
root@slack:/nginx# tar xvf nginx.tar.gz
nginx/
nginx/doinst.sh
nginx/nginx.info
nginx/nginx.logrotate
nginx/slack-desc
nginx/rc.nginx
nginx/README
nginx/nginx.SlackBuild
root@slack:/nginx# ls
nginx/  nginx-1.23.2.tar.gz  nginx.tar.gz
root@slack:/nginx# _
```

Movemos de la carpeta nginx el archivo nginx-1.23.tar.gz

```
root@slack:/nginx# ls
nginx/  nginx-1.23.2.tar.gz  nginx.tar.gz
root@slack:/nginx# pwd
/nginx
root@slack:/nginx# mv nginx-1.23.2.tar.gz /nginx/nginx
root@slack:/nginx# ls
nginx/  nginx.tar.gz
root@slack:/nginx# cd nginx
root@slack:/nginx/nginx# ls
README  nginx-1.23.2.tar.gz  nginx.info  rc.nginx
doinst.sh  nginx.SlackBuild*  nginx.logrotate  slack-desc
root@slack:/nginx/nginx# _
```

Descomprimir el archivo nginx-1.23.2.tar.gz con tar xvf

```
root@slack:/nginx/nginx# tar xvf nginx-1.23.2.tar.gz _
```

```
slackwareweb Clone [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
nginx-1.23.2/auto/lib/pcre/
nginx-1.23.2/auto/lib/perl/
nginx-1.23.2/auto/lib/zlib/
nginx-1.23.2/auto/lib/zlib/conf
nginx-1.23.2/auto/lib/zlib/make
nginx-1.23.2/auto/lib/zlib/makefile.bcc
nginx-1.23.2/auto/lib/zlib/makefile.msuc
nginx-1.23.2/auto/lib/zlib/makefile.ouc
nginx-1.23.2/auto/lib/perl/conf
nginx-1.23.2/auto/lib/perl/make
nginx-1.23.2/auto/lib/pcre/conf
nginx-1.23.2/auto/lib/pcre/make
nginx-1.23.2/auto/lib/pcre/makefile.bcc
nginx-1.23.2/auto/lib/pcre/makefile.msuc
nginx-1.23.2/auto/lib/pcre/makefile.ouc
nginx-1.23.2/auto/lib/openssl/conf
nginx-1.23.2/auto/lib/openssl/make
nginx-1.23.2/auto/lib/openssl/makefile.bcc
nginx-1.23.2/auto/lib/openssl/makefile.msuc
nginx-1.23.2/auto/lib/libbsd/conf
nginx-1.23.2/auto/lib/libatomic/conf
nginx-1.23.2/auto/lib/libatomic/make
nginx-1.23.2/auto/lib/google-perftools/conf
nginx-1.23.2/auto/lib/goip/conf
nginx-1.23.2/auto/cc/acc
nginx-1.23.2/auto/cc/bcc
nginx-1.23.2/auto/cc/cc
nginx-1.23.2/auto/cc/clang
nginx-1.23.2/auto/cc/conf
nginx-1.23.2/auto/cc/gcc
nginx-1.23.2/auto/cc/icc
nginx-1.23.2/auto/cc/msuc
nginx-1.23.2/auto/cc/name
nginx-1.23.2/auto/cc/ouc
nginx-1.23.2/auto/cc/sunc
root@slack:/nginx/nginx#
```

Entrar a la carpeta nginx-1.23.2 que se creó después de descomprimir

```
nginx-1.23.2/auto/cc/sunc
root@slack:/nginx/nginx# ls
README      nginx-1.23.2/      nginx.SlackBuild*  nginx.logrotate  slack-desc
doinst.sh   nginx-1.23.2.tar.gz nginx.info          rc.nginx
root@slack:/nginx/nginx# cd nginx-1.23.2
root@slack:/nginx/nginx/nginx-1.23.2# ls
CHANGES  CHANGES.ru  LICENSE  README  auto/  conf/  configure*  contrib/  html/  man/  src/
root@slack:/nginx/nginx/nginx-1.23.2# ./configure
```

Ejecutamos el configure

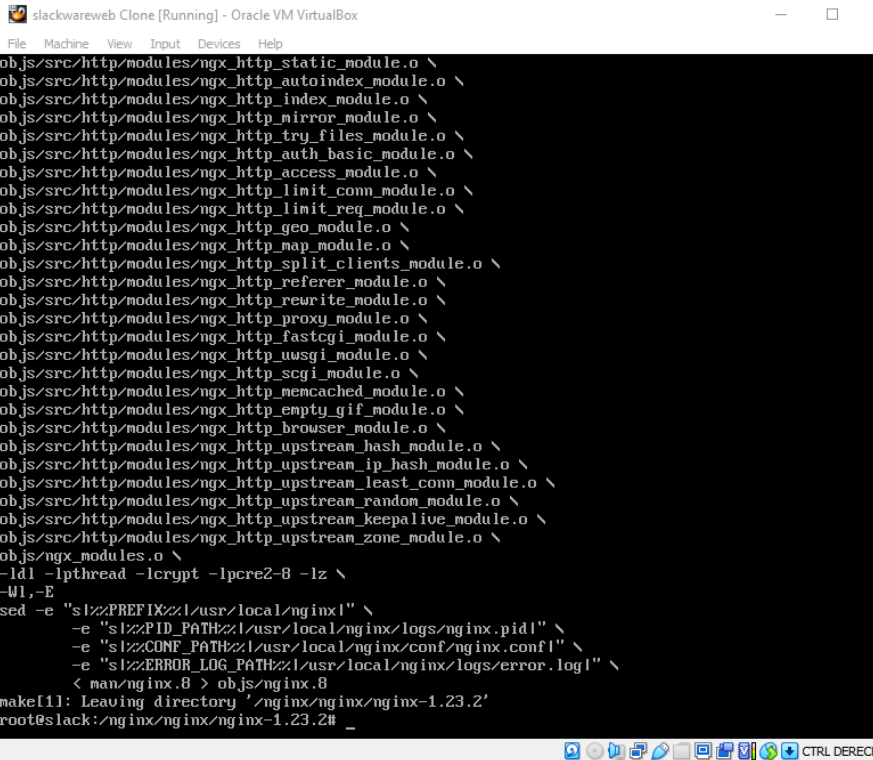
```
slackwareweb Clone [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
checking for System V shared memory ... found
checking for POSIX semaphores ... not found
checking for POSIX semaphores in libpthread ... found
checking for struct msghdr.msghdr ... found
checking for ioctl(FIONBIO) ... found
checking for ioctl(FIONREAD) ... found
checking for struct tm.tm_gmtoff ... found
checking for struct dirent.d_namlen ... not found
checking for struct dirent.d_type ... found
checking for sysconf(_SC_NPROCESSORS_ONLN) ... found
checking for sysconf(_SC_LEVEL1_DCACHE_LINESIZE) ... found
checking for openat(), fstatat() ... found
checking for getaddrinfo() ... found
checking for PCRE2 library ... found
checking for zlib library ... found
creating objs/Makefile

configuration summary
+ using system PCRE2 library
+ OpenSSL library is not used
+ using system zlib library

nginx path prefix: "/usr/local/nginx"
nginx binary file: "/usr/local/nginx/sbin/nginx"
nginx modules path: "/usr/local/nginx/modules"
nginx configuration prefix: "/usr/local/nginx/conf"
nginx configuration file: "/usr/local/nginx/conf/nginx.conf"
nginx pid file: "/usr/local/nginx/logs/nginx.pid"
nginx error log file: "/usr/local/nginx/logs/error.log"
nginx http access log file: "/usr/local/nginx/logs/access.log"
nginx http client request body temporary files: "client_body_temp"
nginx http proxy temporary files: "proxy_temp"
nginx http fastcgi temporary files: "fastcgi_temp"
nginx http uwsgi temporary files: "uwsigi_temp"
nginx http scgi temporary files: "scgi_temp"
root@slack:/nginx/nginx/nginx-1.23.2#
```

Ahora utilizamos el comando make para compilar el código fuente para generar archivos necesarios

```
root@slack:/nginx/nginx/nginx-1.23.2# make_
```



```
objs/src/http/modules/ngx_http_static_module.o \
objs/src/http/modules/ngx_http_autoindex_module.o \
objs/src/http/modules/ngx_http_index_module.o \
objs/src/http/modules/ngx_http_mirror_module.o \
objs/src/http/modules/ngx_http_try_files_module.o \
objs/src/http/modules/ngx_http_auth_basic_module.o \
objs/src/http/modules/ngx_http_access_module.o \
objs/src/http/modules/ngx_http_limit_conn_module.o \
objs/src/http/modules/ngx_http_limit_req_module.o \
objs/src/http/modules/ngx_http_geo_module.o \
objs/src/http/modules/ngx_http_map_module.o \
objs/src/http/modules/ngx_http_split_clients_module.o \
objs/src/http/modules/ngx_http_referer_module.o \
objs/src/http/modules/ngx_http_rewrite_module.o \
objs/src/http/modules/ngx_http_proxy_module.o \
objs/src/http/modules/ngx_http_fastcgi_module.o \
objs/src/http/modules/ngx_http_uwsgi_module.o \
objs/src/http/modules/ngx_http_scgi_module.o \
objs/src/http/modules/ngx_http_memcached_module.o \
objs/src/http/modules/ngx_http_empty_gif_module.o \
objs/src/http/modules/ngx_http_browser_module.o \
objs/src/http/modules/ngx_http_upstream_hash_module.o \
objs/src/http/modules/ngx_http_upstream_ip_hash_module.o \
objs/src/http/modules/ngx_http_upstream_least_conn_module.o \
objs/src/http/modules/ngx_http_upstream_random_module.o \
objs/src/http/modules/ngx_http_upstream_keepalive_module.o \
objs/src/http/modules/ngx_http_upstream_zone_module.o \
objs/nginx_modules.o \
-lld -lpthread -lcrypt -lpcr2-8 -lz \
-Wl,-E
sed -e "s|%%PREFIX%%|usr/local/nginx|" \
-e "s|%%PID_PATH%%|usr/local/nginx/logs/nginx.pid|" \
-e "s|%%CONF_PATH%%|usr/local/nginx/conf/nginx.conf|" \
-e "s|%%ERROR_LOG_PATH%%|usr/local/nginx/logs/error.log|" \
< man/nginx.8 > objs/nginx.8
make[1]: Leaving directory '/nginx/nginx/nginx-1.23.2'
root@slack:/nginx/nginx/nginx-1.23.2# _
```

Hacemos directorios para nginx con make install

```
root@slack:/nginx/nginx/nginx-1.23.2# make install_
```



```
slackwareweb Clone [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
cp objs/nginx /usr/local/nginx/sbin/nginx
test -d /usr/local/nginx/conf \
  || mkdir -p /usr/local/nginx/conf
cp conf/koi-win /usr/local/nginx/conf
cp conf/koi-utf /usr/local/nginx/conf
cp conf/win-utf /usr/local/nginx/conf
test -f /usr/local/nginx/conf/mime.types \
  || cp conf/mime.types /usr/local/nginx/conf
cp conf/mime.types /usr/local/nginx/conf/mime.types.default
test -f /usr/local/nginx/conf/fastcgi_params \
  || cp conf/fastcgi_params /usr/local/nginx/conf
cp conf/fastcgi_params \
  /usr/local/nginx/conf/fastcgi_params.default
test -f /usr/local/nginx/conf/fastcgi.conf \
  || cp conf/fastcgi.conf /usr/local/nginx/conf
cp conf/fastcgi.conf /usr/local/nginx/conf/fastcgi.conf.default
test -f /usr/local/nginx/conf/uwsgi_params \
  || cp conf/uwsgi_params /usr/local/nginx/conf
cp conf/uwsgi_params \
  /usr/local/nginx/conf/uwsgi_params.default
test -f /usr/local/nginx/conf/scgi_params \
  || cp conf/scgi_params /usr/local/nginx/conf
cp conf/scgi_params \
  /usr/local/nginx/conf/scgi_params.default
test -f /usr/local/nginx/conf/nginx.conf \
  || cp conf/nginx.conf /usr/local/nginx/conf/nginx.conf
cp conf/nginx.conf /usr/local/nginx/conf/nginx.conf.default
test -d /usr/local/nginx/logs \
  || mkdir -p /usr/local/nginx/logs
test -d /usr/local/nginx/logs \
  || mkdir -p /usr/local/nginx/logs
test -d /usr/local/nginx/html \
  || cp -R html /usr/local/nginx
test -d /usr/local/nginx/logs \
  || mkdir -p /usr/local/nginx/logs
make[1]: Leaving directory /nginx/nginx/nginx-1.23.2
root@slack:/usr/local/nginx/nginx-1.23.2#
```

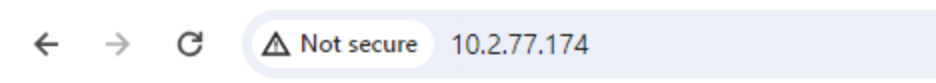
Iniciamos el servicio de nginx

```
root@slack:/usr/local/nginx/sbin/nginx
```

Realizamos la página web en la carpeta index.html

```
root@slack:/usr/local/nginx/html/index.html
```

Creamos la página web



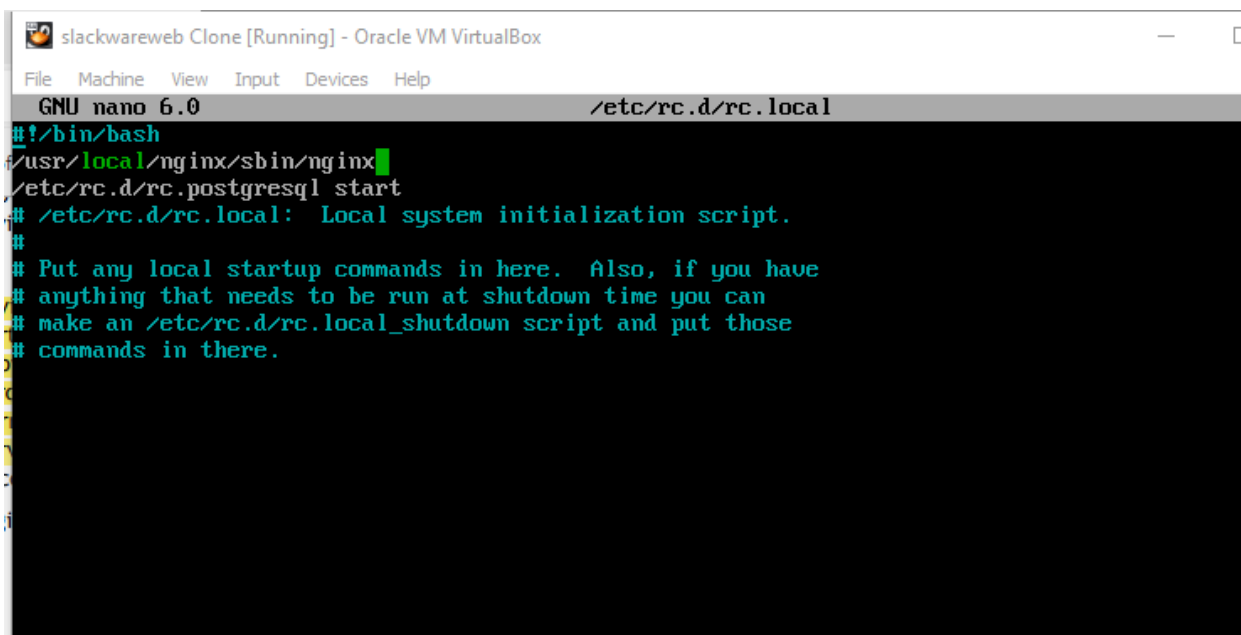
Laboratorio 6 de RECO!

Se hizo lo que se pudo con esta pagina web,admiren a BTS.



Derechos de autor de Sofia XD

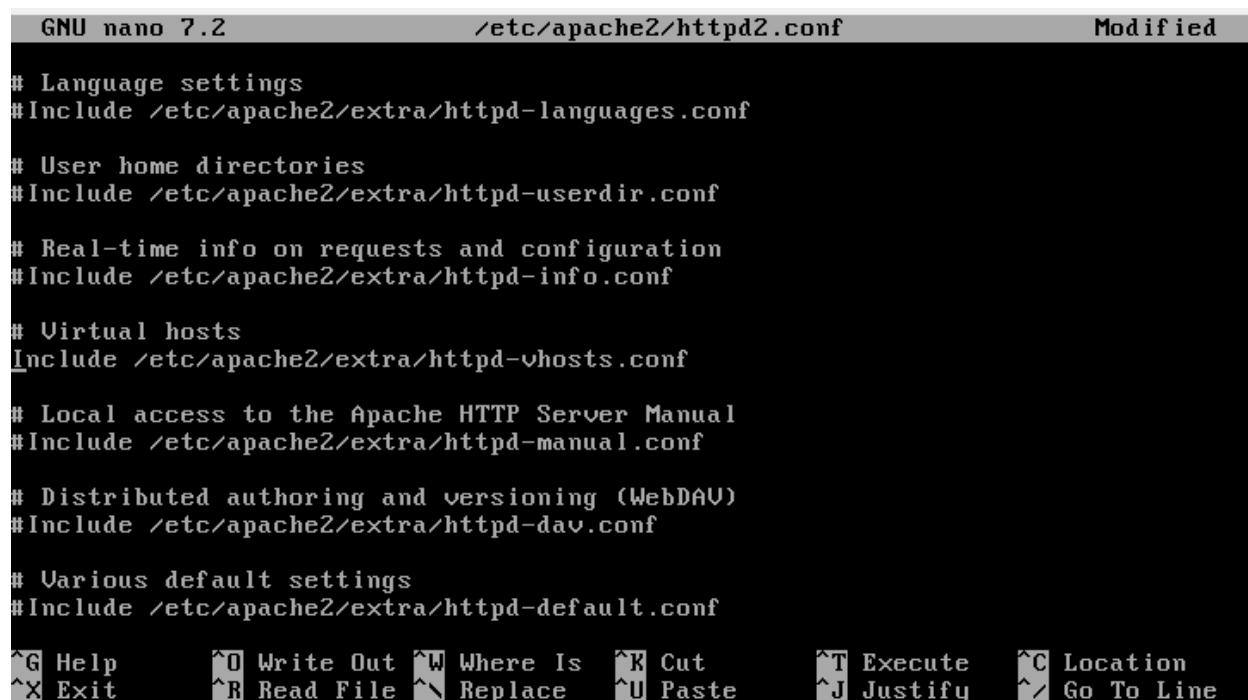
Para que arranque solo



```
slackwareweb Clone [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
GNU nano 6.0 /etc/rc.d/rc.local
#!/bin/bash
/usr/local/nginx/sbin/nginx
/etc/rc.d/rc.postgresql start
# /etc/rc.d/rc.local: Local system initialization script.
#
# Put any local startup commands in here. Also, if you have
# anything that needs to be run at shutdown time you can
# make an /etc/rc.d/rc.local_shutdown script and put those
# commands in there.
```

2.2.2.2 Configuración de servicio de hosting

Configuamos el archivo `/etc/apache2/httpd2.conf` activando el servicio de host virtual



```
GNU nano 7.2 /etc/apache2/httpd2.conf Modified
# Language settings
#Include /etc/apache2/extra/httpd-languages.conf

# User home directories
#Include /etc/apache2/extra/httpd-userdir.conf

# Real-time info on requests and configuration
#Include /etc/apache2/extra/httpd-info.conf

# Virtual hosts
Include /etc/apache2/extra/httpd-vhosts.conf

# Local access to the Apache HTTP Server Manual
#Include /etc/apache2/extra/httpd-manual.conf

# Distributed authoring and versioning (WebDAV)
#Include /etc/apache2/extra/httpd-dav.conf

# Various default settings
#Include /etc/apache2/extra/httpd-default.conf

^G Help      ^O Write Out ^W Where Is  ^K Cut       ^T Execute  ^C Location
^X Exit      ^R Read File ^\ Replace   ^U Paste     ^J Justify  ^_ Go To Line
```

Indicamos los hosts virtuales como se nos pide en la guía

```
GNU nano 7.2 /etc/apache2/extra/httpd-vhosts.conf
DocumentRoot "/var/www/docs/dummy-host2.example.com"
ServerName dummy-host2.example.com
ErrorLog "logs/dummy-host2.example.com-error_log"
CustomLog "logs/dummy-host2.example.com-access_log" common
</VirtualHost>

<VirtualHost *:80>
    DocumentRoot "/var/www/htdocs/Redes"
    ServerName redes.milo.com.co
</VirtualHost>

<VirtualHost *:80>
    DocumentRoot "/var/www/htdocs/Seguridad"
    ServerName seguridad.sofia.org.jp
</VirtualHost>

<VirtualHost *:80>
    DocumentRoot "/var/www/htdocs/Sistemas"
    ServerName sistemas.milo.com.cl
</VirtualHost>

OPENF#
```

Creamos las carpetas previamente indicadas y les creamos su index.html correspondientes

```
OPENF# /etc/rc.d/apache2 restart
apache2(ok)
apache2(ok)
OPENF# nslookup
> www.gil.com
Server:      ::1
Address:     ::1#53

Name:   www.gil.com
Address: 10.2.77.174
> dns.gil.com
Server:      ::1
Address:     ::1#53

Name:   dns.gil.com
Address: 10.2.77.172
> exit

OPENF# mkdir /var/www/htdocs/Redes
OPENF# mkdir /var/www/htdocs/Seguridad
OPENF# mkdir /var/www/htdocs/Sistemas
OPENF# touch /var/www/htdocs/Sistemas/index.html
OPENF# touch /var/www/htdocs/Redes/index.html
OPENF# touch /var/www/htdocs/Seguridad/index.html
OPENF#
```

Creamos una zona para cada dominio.

```
GNU nano 7.2      named.conf      Modified
};
zone "gil.com"{
    type Master;
    file "gil.com.hosts";
};

zone "milo.com.co"{
    type Master;
    file "milo.com.co.hosts";
};

zone "sofia.org.jp"{
    type Master;
    file "sofia.org.jp.hosts";
};

zone "milo.com.cl"{
    type Master;
    file "milo.com.cl.hosts";
};

^G Help      ^O Write Out ^W Where Is  ^K Cut       ^T Execute   ^C Location
^X Exit      ^R Read File ^_ Replace   ^U Paste     ^J Justify   ^_ Go To Line
```

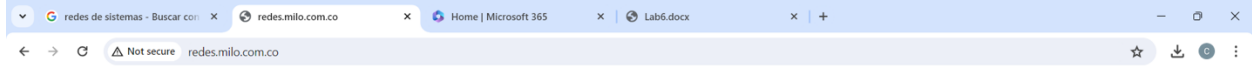
nds

Configuramos los archivos .hosts correspondientes a cada zona

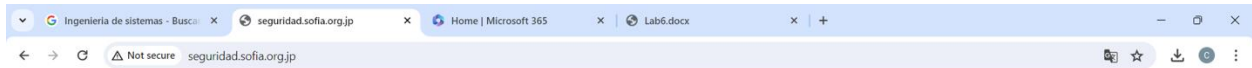
```
Before reporting a bug, please try to reproduce it with the latest
version of the code. With bug reports, please try to ensure that
enough information to reproduce the problem is enclosed, and if a
known fix for it exists, include that as well.

You have new mail.
OPENF# cd /var/named/etc/DNS
OPENF# ls
castano.com.it.hosts      managed-keys.bind        milo.com.co.hosts
gil.com.hosts            managed-keys.bind.jnl    named.ca
gil.org.uk.hosts         milo.com.cl.hosts        sofia.org.jp.hosts
OPENF# kbd us
kbd: keyboard mapping set to us
fia.org.jp.hosts sofia.org.jp.hosts <
ksh: check-zone: not found
one sofia.org.jp sofia.org.jp.hosts <
zone sofia.org.jp/IN: loaded serial 2024021902
OK
OPENF# named-checkzone milo.com.co milo.com.co.hosts
zone milo.com.co/IN: loaded serial 2024021902
OK
checkzone milo.com.cl milo.com.cl.hosts <
zone milo.com.cl/IN: loaded serial 2024021902
OK
OPENF#
```

Confirmamos que se pueden visualizar todas las paginas

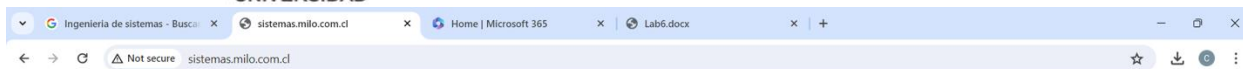


Pagina Redes



Pagina Seguridad





Pagina Sistemas



3. Conclusiones

La infraestructura tecnológica detrás de los servicios web y el hosting es compleja y abarca una variedad de elementos, desde estaciones de usuario hasta servidores físicos y virtualizados, switches, routers y servicios en la nube. Esta diversidad de componentes permite ofrecer entornos completos y seguros para nuestra sustentación de red.

La configuración de hosting en OpenBSD con Apache implica la instalación y configuración de Apache para alojar diferentes tipos de contenido web, como páginas estáticas, la utilización de DNS.

Gracias a Packet Tracer vimos la simulación de una infraestructura de los servidores, realizando la configuración de los routers de manera que las rutas sean estáticas, viendo la conectividad de los PC de la misma LAN y hacia diferentes redes, utilizando el protocolo ICMP para verificar la comunicación entre los PCs y montando la conectividad multiuser.

Completamos la instalación de los servicios web en cada una de las maquinas virtuales como Slackware, OpenBDS y WindowsServer, realizando satisfactoriamente las paginas web y haciendo la automatización de lo servicios.

4. Bibliografía

- <https://www.ibm.com/docs/es/was/9.0.5?topic=services-web>
- <https://www.youtube.com/watch?v=9xMOaKLqPIo>

