## 

## 

## 

## 

## Instalacion y Configuracion Router en Ubuntu Server

## Marcos Alonso Cabral, ASIR2

## 

**Índice**

I[nstalación y configuración de Linux Server con dos adaptadores de red (Adaptador puente y red interna):](#_kjtfzwahslbg)

[- Instalar el servicio DHCP](#_5m9ilt3p37s2)

[- Configurar el servicio DHCP para suministrar la red interna](#_to1cd6k9m1ip)

[- Instalar el servicio DNS](#_3sfmtm9x7zb8)

[- Configurar el servicio DNS para suministrar el servicio a la red interna](#_z233o25z7992)

[- Configurar la tabla de enrutado para que las peticiones se reenvíen de la red interna a la externa y tener acceso a internet](#_k9zktxectmo3)

[- Utilizando un cliente con una sola interfaz en red interna:](#_4js09379o891)

[- Comprobar que se asigna la IP dentro de la subred suministrada](#_tbvfisqyu8zv)

[- Comprobar que se puede hacer ping a ips externas](#_4g76rlc2ioa1)

[- Comprobar que se puede hacer ping y navegar a urls externas](#_q4h2n1abq2b4)

[- Comprobar que se puede acceder a un servicio interno (Por ejemplo, un servidor web Apache)](#_vu6ok1b6mapq)

## 

## Instalación y configuración de Linux Server con dos adaptadores de red (Adaptador puente y red interna):

Primero procedemos a instalar el servidor ubuntu, y como es bastante simple voy a ir dejando capturas seguido de una breve descripción.En esta primera captura seleccionamos el lenguaje:

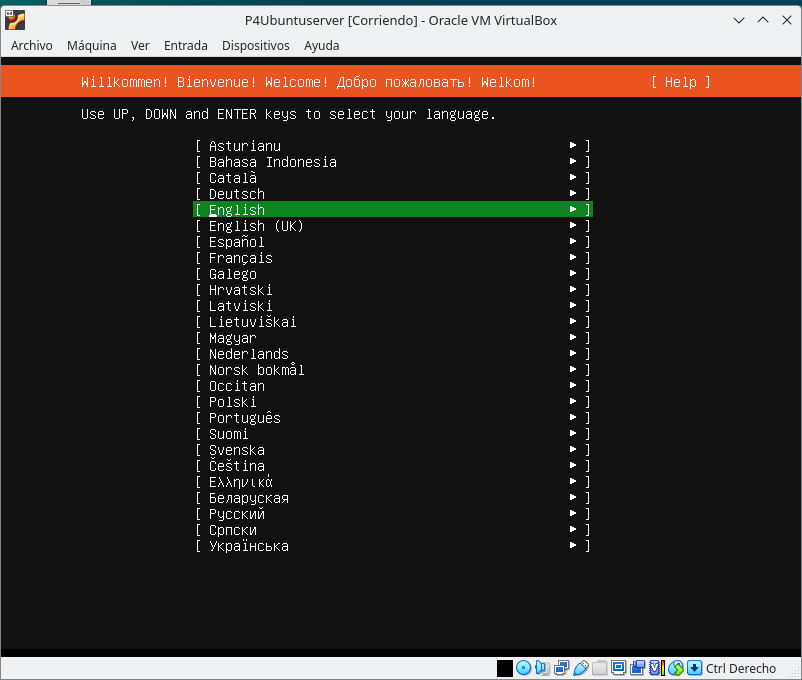


Imagen ilustrativa

Luego con el internet deshabilitado procedemos a instalar el server al completo:

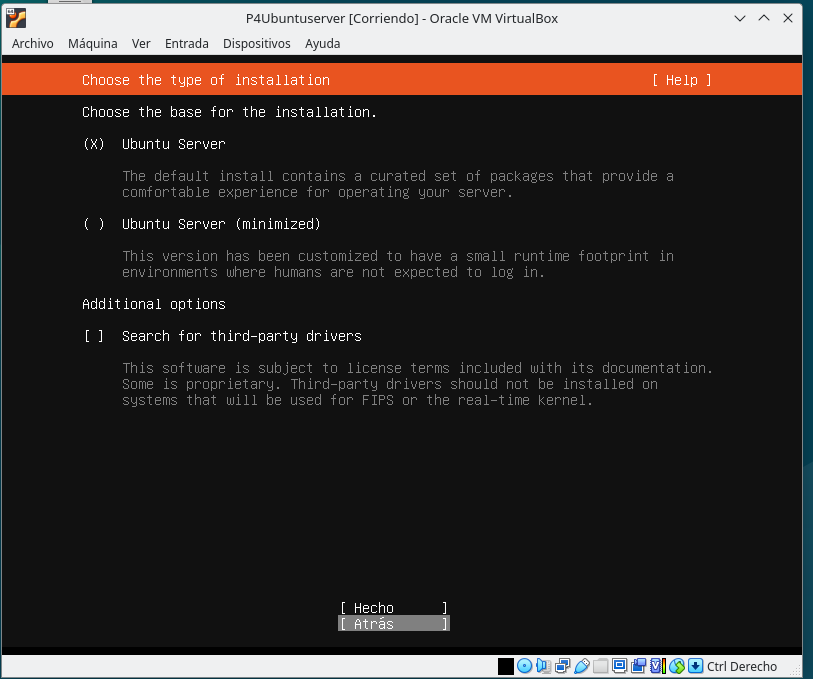


Imagen ilustrativa

Una vez instalado le damos a Hecho:

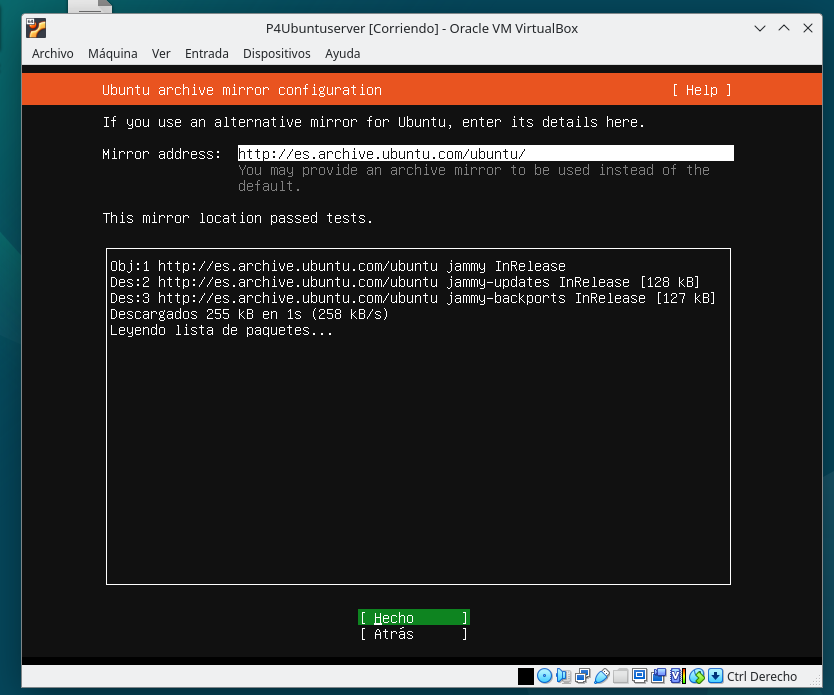


Imagen ilustrativa

Luego le damos a que use todo el disco:

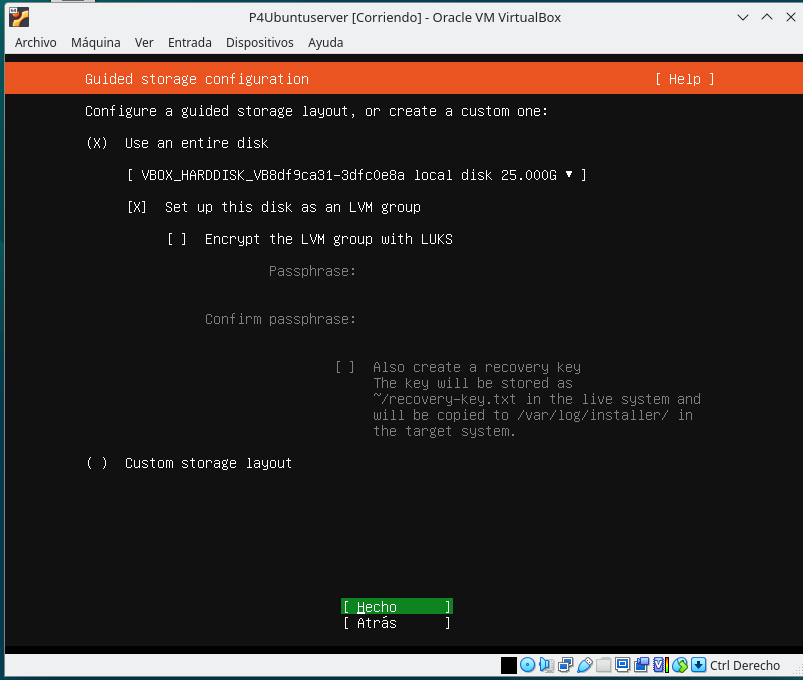


Imagen ilustrativa

Comprobamos la configuración de instalación y le damos a Hecho :

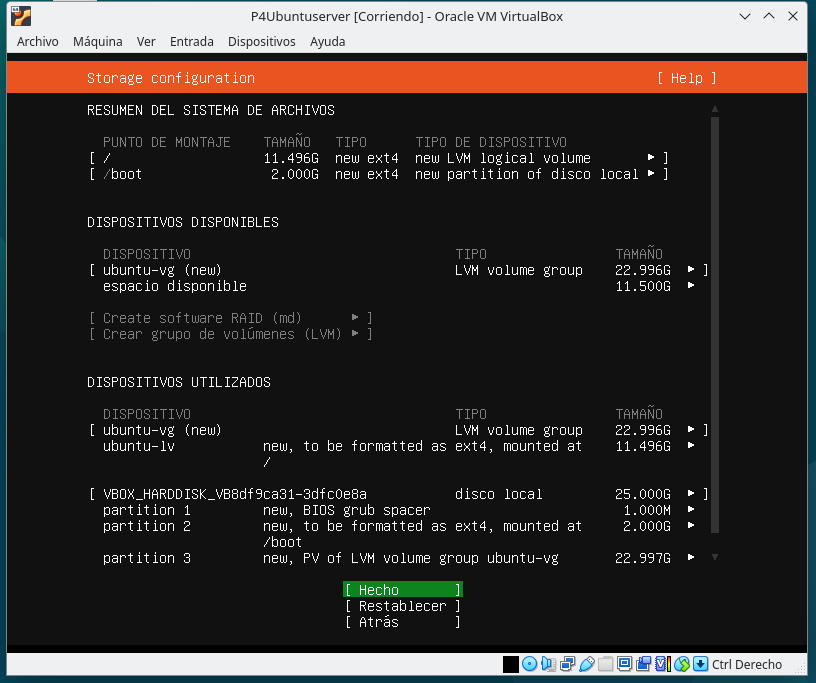


Imagen ilustrativa

Posteriormente procedemos a poner el nombre del server y una contraseña:

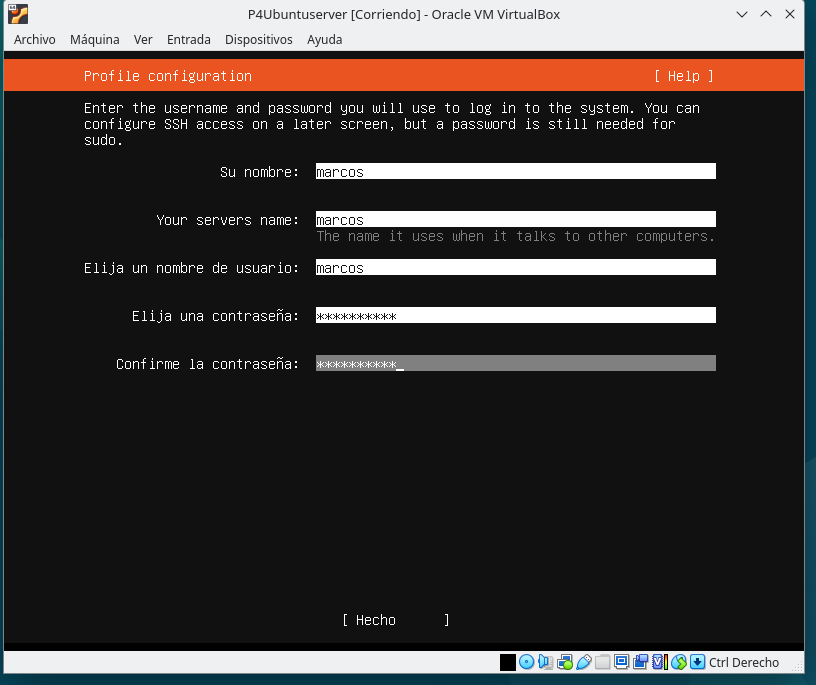


Imagen ilustrativa

Luego saltamos la opción de ubuntu pro:

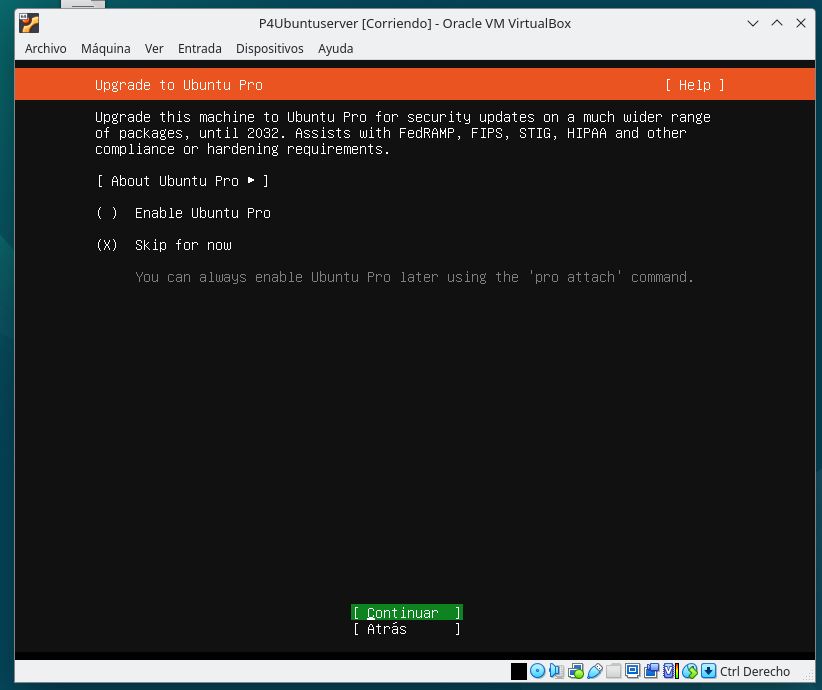


Imagen ilustrativa

Continuamos sin instalar el servidor ssh:

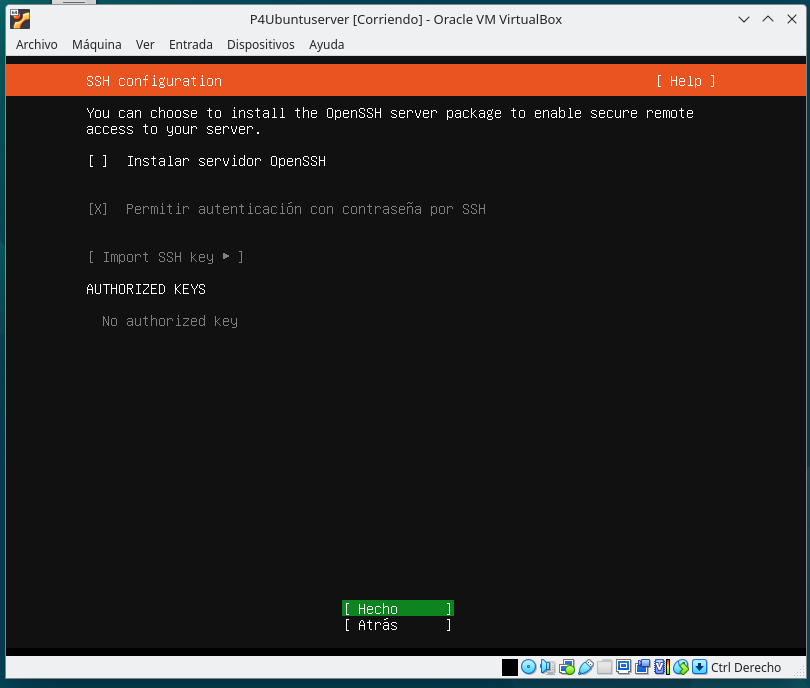


Imagen ilustrativa

Posteriormente procedemos a continuar sin instalar nada a mayores:

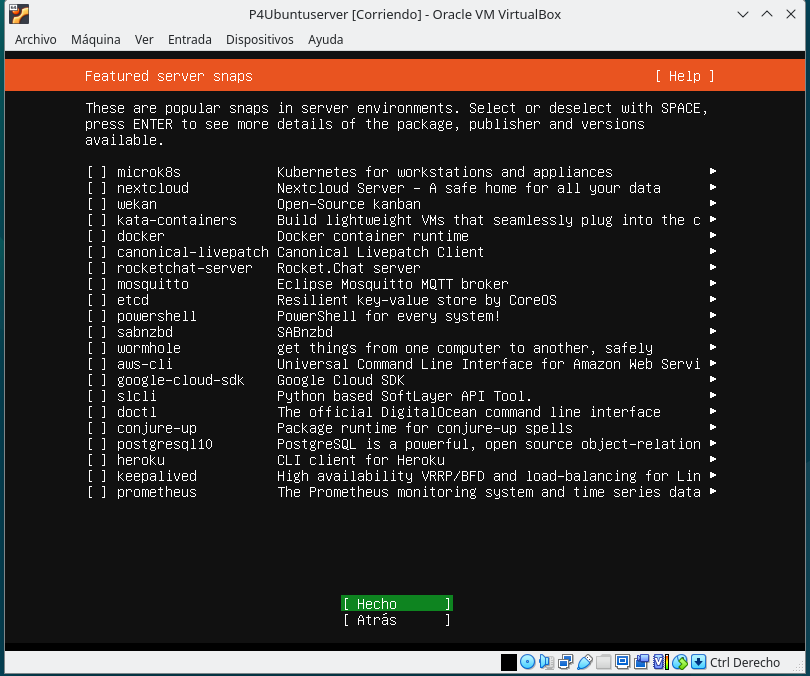


Imagen ilustrativa

Y una vez instalado todo le damos a reiniciar ahora para completar la instalación:

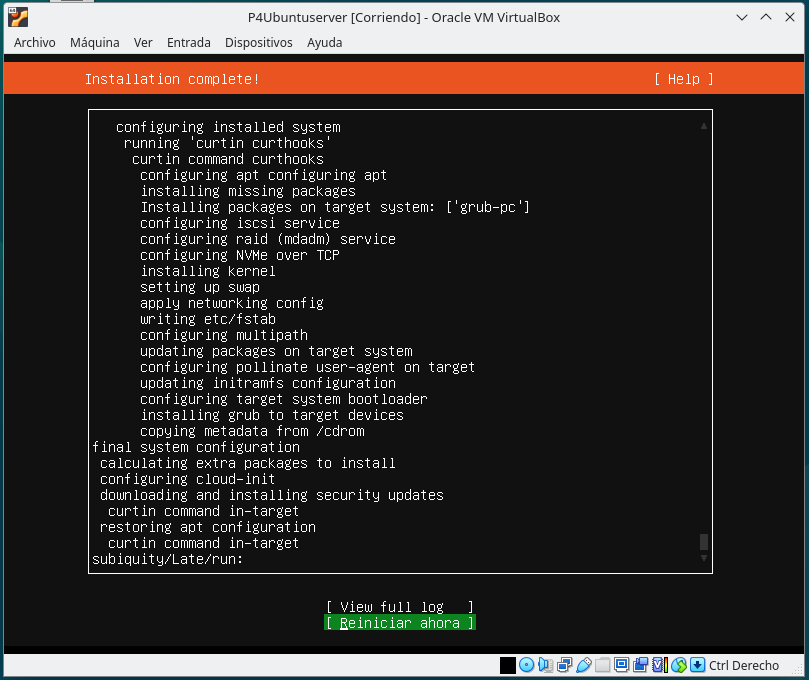


Imagen ilustrativa

## 

## - Instalar el servicio DHCP

Para instalar el servicio simplemente debemos poner el comando “sudo apt install isc-dhcp-server” aunque puede que haga falta hacer un “sudo apt update” anteriormente.

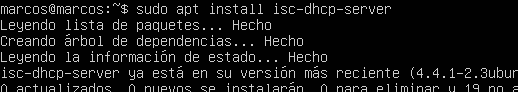


Imagen ilustrativa

## - Configurar el servicio DHCP para suministrar la red interna

Para poder suministrar una red interna, primero debemos añadir un adaptador de red para esa red interna, para ello primero, con la máquina apagada debemos ponemos ir a las opciones de esta máquina y poner red interna para luego permitir todo en el modo promiscuo.

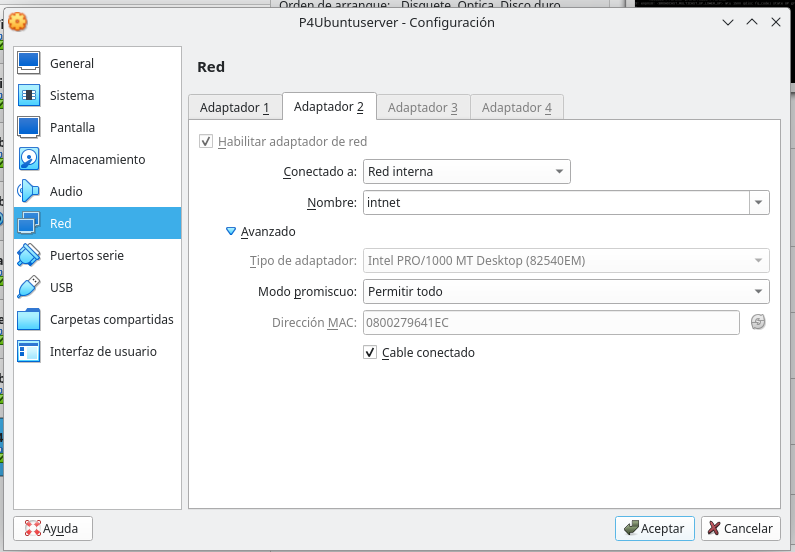


Imagen ilustrativa

Luego debemos modificar el fichero de netplan para configurar estas dos interfaces de red, tanto la que tiene conexión con el exterior como la de red interna. Para ello debemos hacer un “sudo nano /etc/netplan/50-cloud-init.yaml” (si no tienes este fichero tienes que hacer un “sudo netplan generate”) y configurar el adaptador de red enp0s3 y enp0s8 de la siguiente manera (sabiendo que enp0s3 es la interfaz que tiene acceso a internet y la enp0s8 la de la red interna):

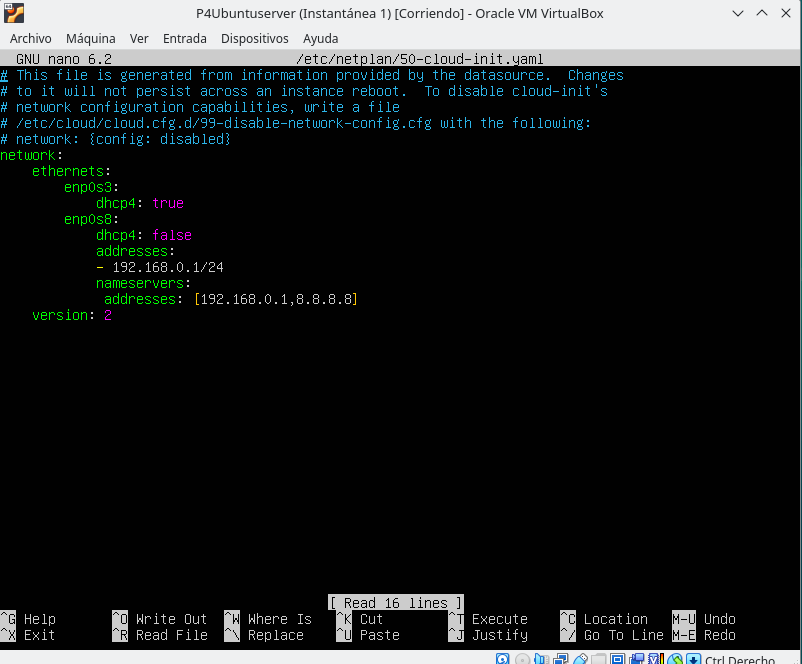


Imagen ilustrativa

Una vez configurado este fichero debemos hacer un “sudo netplan apply” para que se nos apliquen los cambios. (Una vez configurado este fichero .yaml yo recomiendo hacer el comando “sudo cp /etc/netplan/50-cloud-init.yaml /etc/netplan/50-cloud-init.yaml.bckp” ya que así hacemos una copia de seguridad porque justamente este sistema ubuntu tiene un bug que hace que cuando reinicias la maquina no se guarda el netplan que acabamos de configurar.) Y para poder ver que realmente se han realizado estos cambios podemos hacer un “ip a” para ver como nos suministra ambas interfaces:

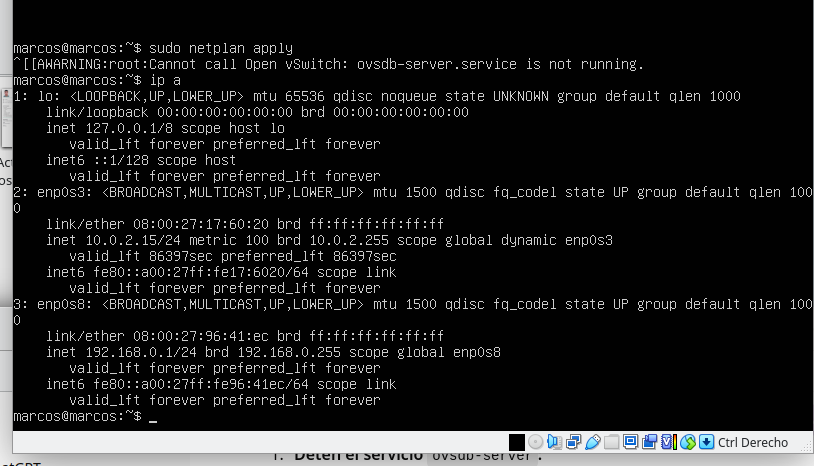


Imagen ilustrativa

Para indicarle cual interfaz va a ser el servidor dhcp (enp0s8) utilizaremos el comando “sudo nano /etc/default/isc-dhcp-server” y en la parte de ipv4 ponemos la interfaz que usaremos que será la enp0s8 (ya que no vamos a utilizar la ipv6).

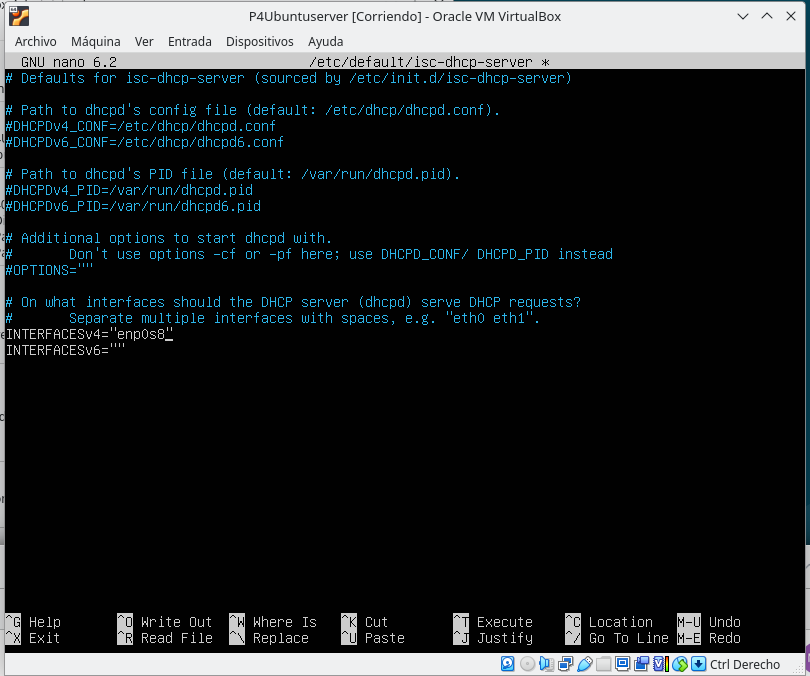


Imagen ilustrativa

Luego para configurar este servicio de dhcp para que sea el servicio dhcp principal primero debemos hacer “sudo nano /etc/dhcp/dhcpd.conf” y borramos almohadilla en “#authoritative”.

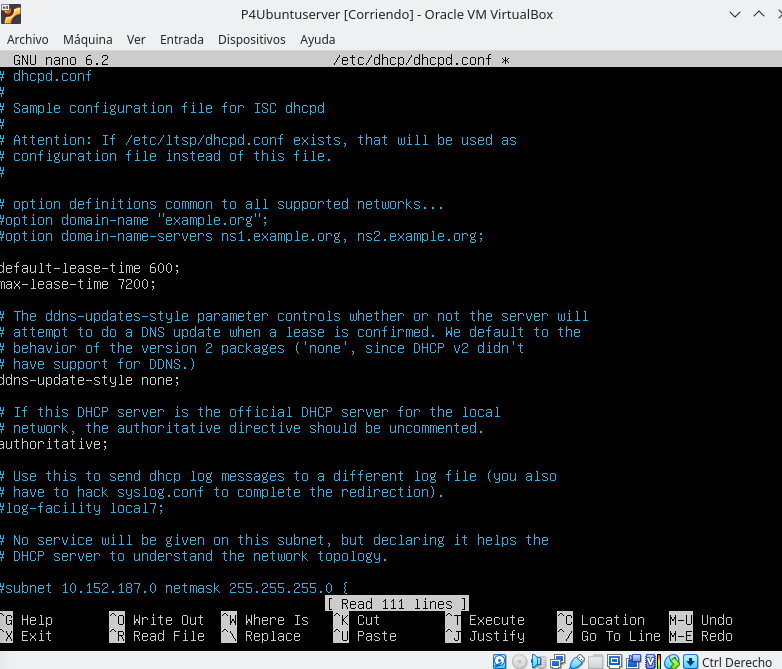


Imagen ilustrativa

Ya que estamos aquí, bajamos en este mismo documento para configurar la subred así como el nombre de dominio del servidor como se puede ver a continuación:

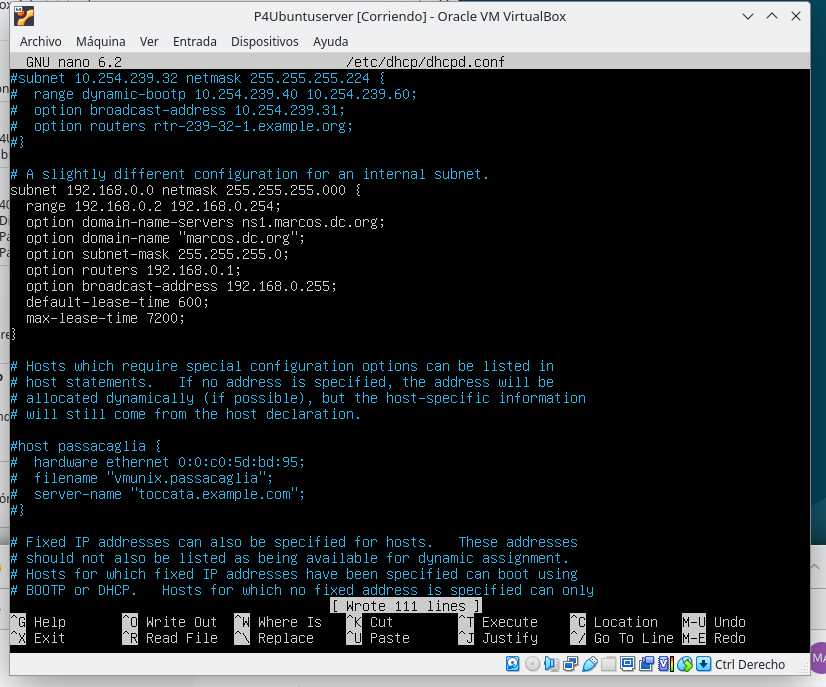


Imagen ilustrativa

Una vez configurado esta subred, debemos escribir las siguientes zonas, que básicamente es la zona del nombre de dominio principal y la negativa de este mismo:

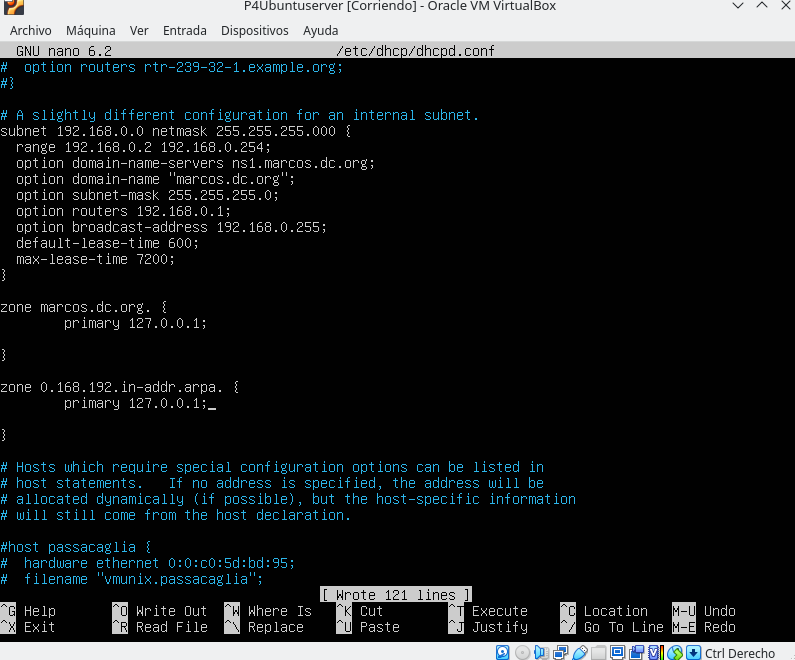


Imagen ilustrativa

Cuando acabemos de configurar y guardar este fichero, debemos salir y poner el comando “sudo ip link set enp0s8 up” .Aunque esto solamente inicia la interfaz de enp0s8 por lo que se puede hacer anteriormente. Solo hay que tener en cuenta que para instalar cosas en el servidor o realizar acciones que necesiten de conexión a internet debemos hacer “sudo ip link set enp0s3 up” o “ “sudo ip link set enp0s8 down” para que se active el la interfaz que tiene conexión a internet (enp0s3) y se desconecte la de la red interna (enp0s8).

Para comprobar que nuestro servicio de dhcp funciona correctamente primero debemos hacer un “sudo service isc-dhcp-server restart” para reiniciar el servicio y luego un “sudo service isc-dhcp-server status” para ver el estado:

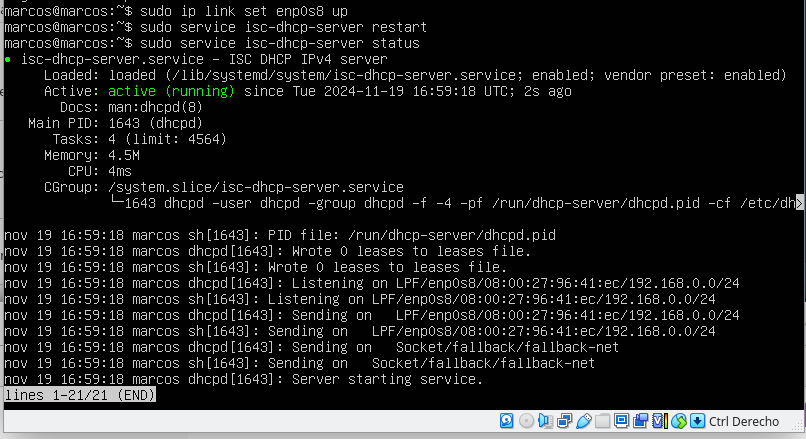


Imagen ilustrativa

## - Instalar el servicio DNS

Antes de nada, debemos desactivar la interfaz de red interna que acabamos de iniciar. Se puede hacer por el comando que dije anteriormente o mediante clic derecho en la parte inferior de la máquina virtual como se puede ver a continuación:

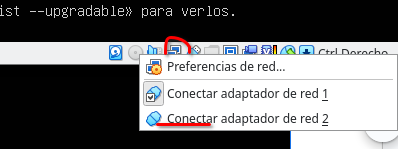


Imagen ilustrativa

Luego ya podemos hacer el “sudo apt update” si es necesario. Y como podemos ver en la siguiente imagen al principio intentaba hacer el update con la interfaz interna activada y luego una vez desactivada ya instalaba:

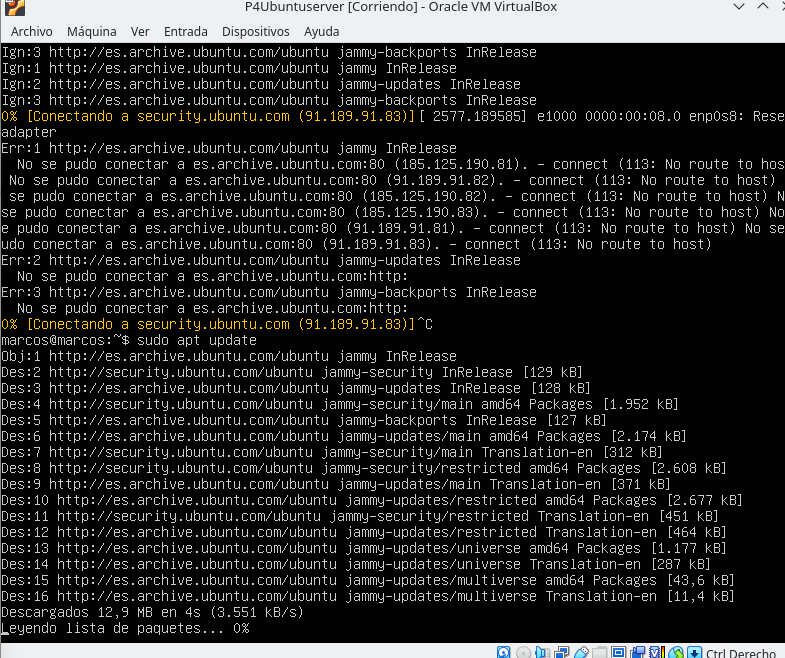


Imagen ilustrativa

Una vez actualizados todos los repositorios instalamos el DNS de bind9 con el comando “sudo apt install bind9 bind9utils bind9-doc”

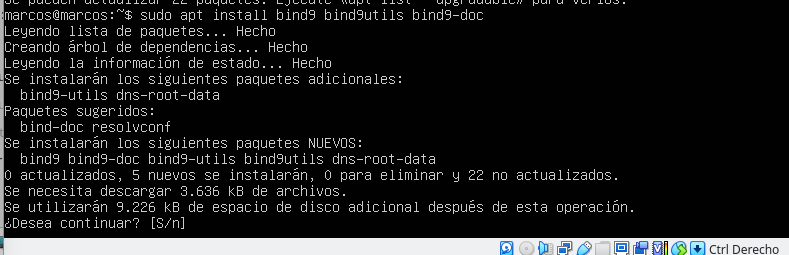


Imagen ilustrativa

## - Configurar el servicio DNS para suministrar el servicio a la red interna

Para empezar a configurar este servicio de DNS debemos acceder al fichero de hosts con “sudo nano /etc/hosts” para así cambiar la primera línea que pone localhost por tu servidor para que así se resuelva por el dns del mismo (que es nombre que le dimos en el documento /etc/dhcp/dhcpd.conf en el apartado de option domain-name-servers):

## 

Imagen ilustrativa

Un fichero que en esta práctica no tocaremos pero que se puede modificar para incluir únicamente las redes que podemos escuchar sería el fichero /etc/bind/named.conf (para modificarlo: “sudo nano /etc/bind/named.conf”).

Luego nos metemos en el fichero /etc/bind/named.conf.local (para acceder: “sudo nano /etc/bind/named.conf.local”) y ponemos la zona de nuestro servidor junto con su zona negativa que también configuramos en el servicio dhcp.

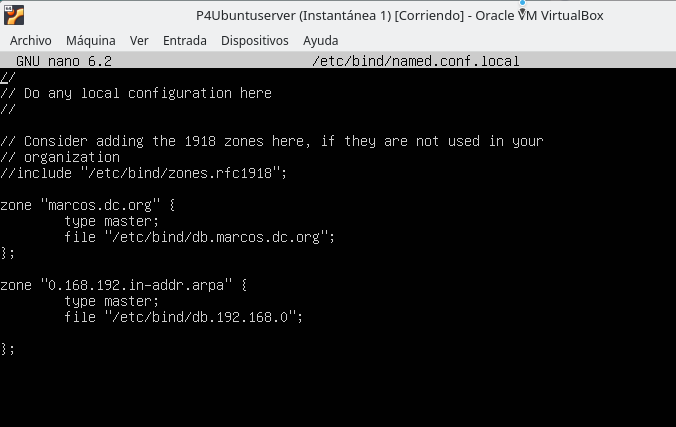


Imagen ilustrativa

Luego copiamos el archivo de base de datos de la zona local para luego modificarla con nuestro dominio del servidor:



Imagen ilustrativa

Posteriormente hacemos un sudo nano a este fichero que acabamos de copiar (etc/bind/db.marcos.dc.org)

Y aquí modificamos todos los parámetros poniendo los de nuestro servidor. Cambiamos la parte de la A que es la ipv4 y eliminamos la AAAA que es la ipv6 ya que no vamos a recibir peticiones ipv6.

Una cosa importante es poner puntos al final de la dirección de nuestro servidor ya que sino da error. Una vez acabada la configuración quedaría tal que así:

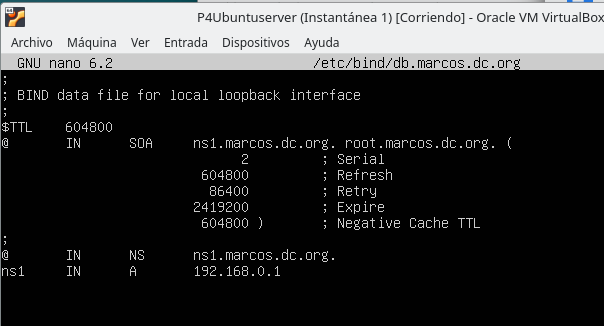


Imagen ilustrativa

A continuación volvemos a copiar el fichero pero esta vez como es de la parte negativa del dominio hacemos el comando “sudo cp /etc/bind/db.127 /etc/bind/db.192.168”



Imagen ilustrativa

Luego accedemos a ese fichero que acabamos de modificar con “sudo nano /etc/bind/db.192.168.0” y cambiamos lo mismo, y para que se vea todo lo que modificó gráficamente, voy a dejar una imagen del archivo sin modificar y a continuación el ya modificado y listo para guardar (como se puede apreciar cambiamos todo lo que poner localhost por nuestro nombre de dominio de servidor):

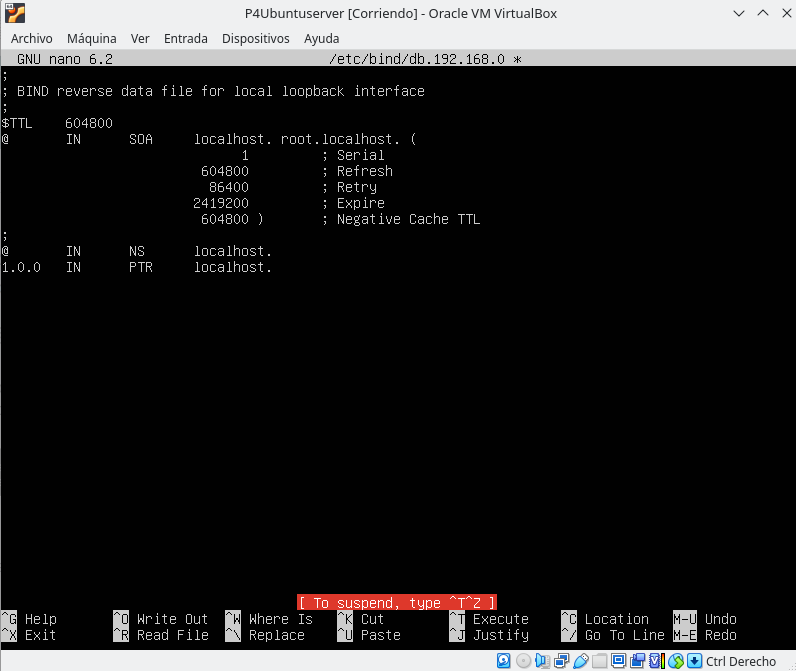


Imagen ilustrativa

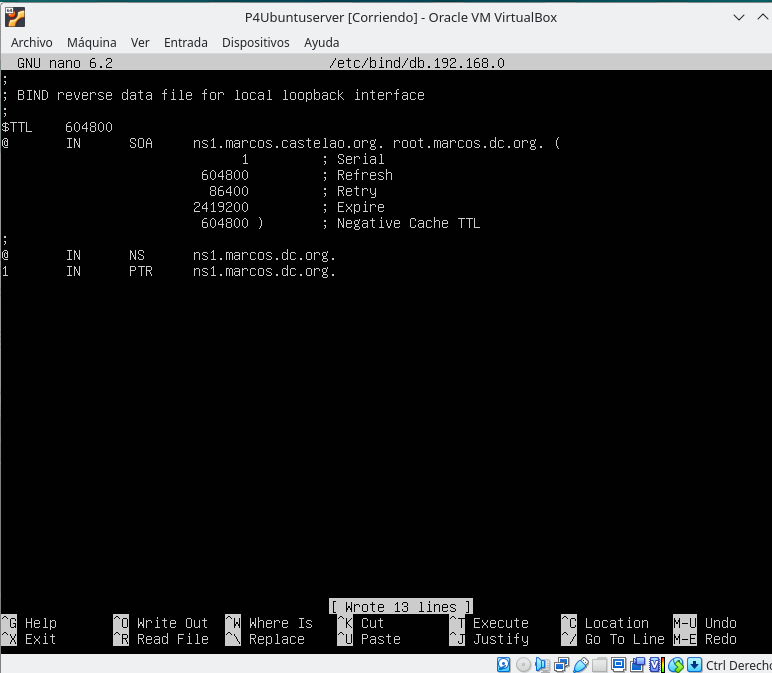


Imagen ilustrativa

Una vez configurado esto, si esta todo bien configurado, al hacer un “sudo named-checkconf” debería de no aparecer nada:



Imagen ilustrativa

Luego para comprobar el otro fichero de configuración hacemos un “sudo named-checkconf /etc/bind/named.conf” y debería de dar lo mismo:



Imagen ilustrativa

Luego para ver si las zonas están bien configuradas ponemos lo siguiente con “checkzone”, primero un “sudo named-chechzone marcos.dc.org /etc/bind/db.marcos.dc.org” para ver si la sintaxis de la zona del nombre de dominio del servidor está correcta y luego un “sudo named-checkzone 0.168.192.in-addr.arpa /etc/bin/db.192.168.0” para comprobar si la sintaxis de la zona negativa también está correcta (si no esta correcto, evidentemente saltara un error) :

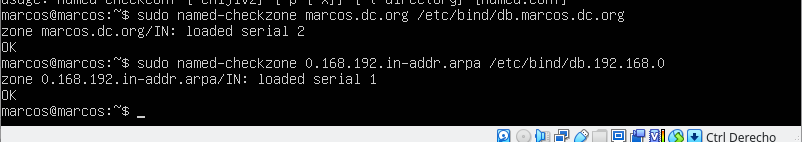


Imagen ilustrativa

Luego reiniciamos el servicio con “sudo service named restart” y vemos el status su estado (está activo en mi caso):

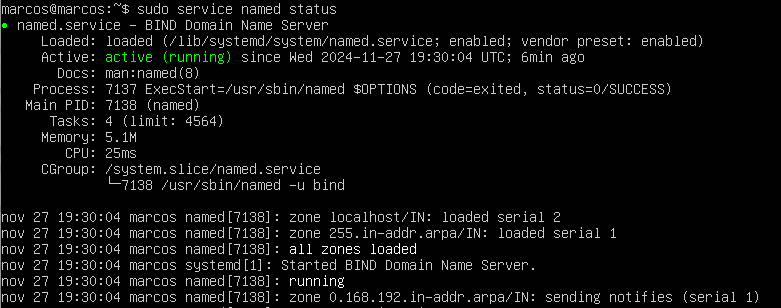


Imagen ilustrativa

Luego en el fichero “/etc/bind/named.conf.options” modificamos el contenido con “sudo nano /etc/bind/named.conf.options” y ponemos los forwarders que seran los servidores a los que enviamos las peticiones (8.8.8.8 google) si no somos capaces de responderlas por nosotros mismos y cambiamos el “dnssec-validation” en ”no” para que luego nos deje conectarnos y no nos bloquee la conexión.

A continuación volveré a dejar una image del fichero sin modificar seguido del fichero modificado :

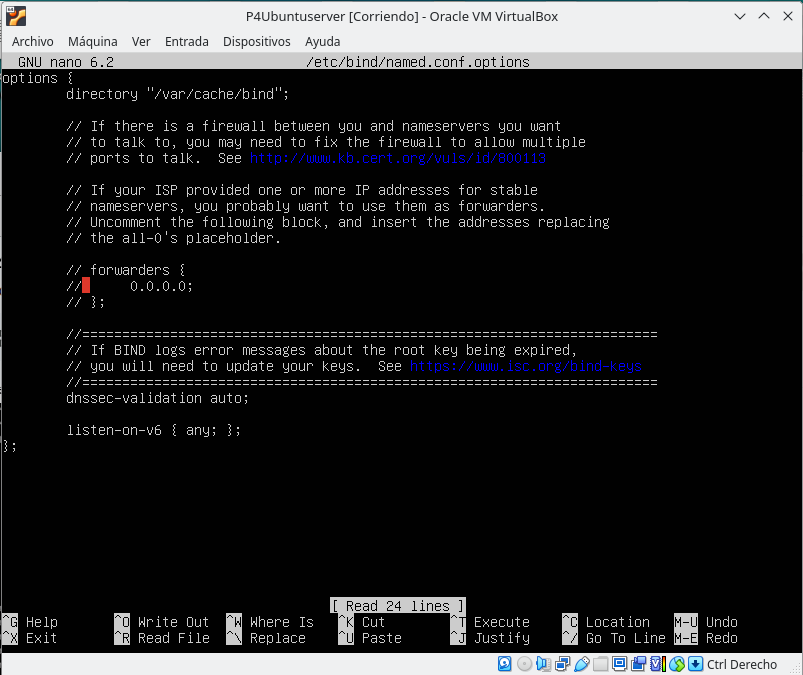


Imagen ilustrativa

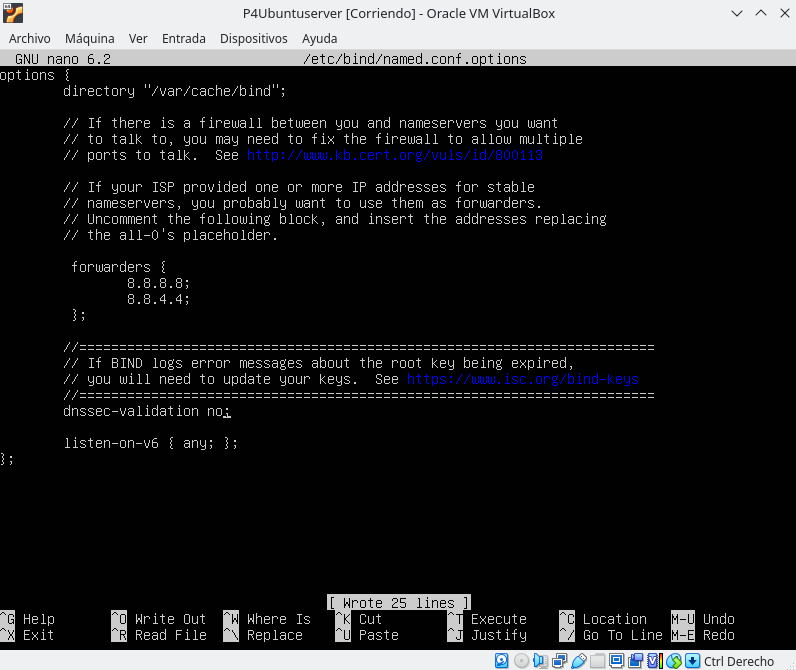


Imagen ilustrativa

Una vez configurado todo voy a comprobar que al menos tiene conexión el cliente para google (con el comando “nslookup google.es” y cabe destacar que el cliente debe estar en red interna). Ya que nslookup pregunta a nuestro servidor DNS que si tiene la ip de google que es la única que tiene junto con 8.8.4.4 (que también es google) por eso puede resolver la consulta. Y para resolver otras consultas necesitamos la tabla de enrutado que haremos a continuación:

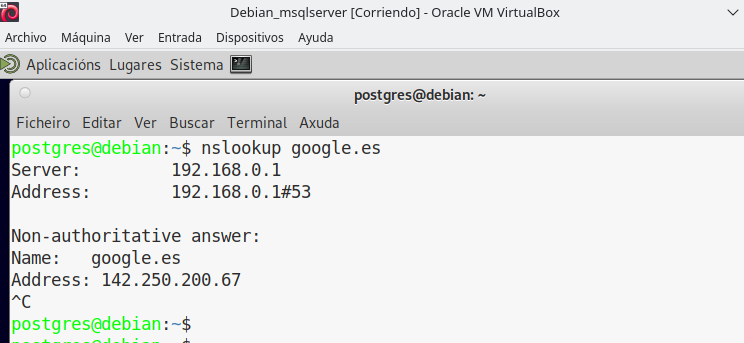


Imagen ilustrativa

## - Configurar la tabla de enrutado para que las peticiones se reenvíen de la red interna a la externa y tener acceso a internet

Para que el cliente pueda resolver otras consultas externas necesitamos cambiar un ajuste del fichero /etc/dhcp/dhcpd.conf, cambiando la “option domain-name-servers” por la ip del servidor (192.168.0.1) y la ip del servidor de google (8.8.8.8). La razón de poner como segunda opción la ip de google es para que el cliente al acceder a internet primero resuelva por nuestro servidor dns y tenga a google de servidor backup y que la conexion sea mas rapido al no tener que estar preguntando nuestro servidor principal por todas las peticiones que no sepa responder (así las responde google).

A continuación dejaré las capturas de el fichero “/etc/dhcp/dhcpd.conf” sin modificar y luego el modificado:

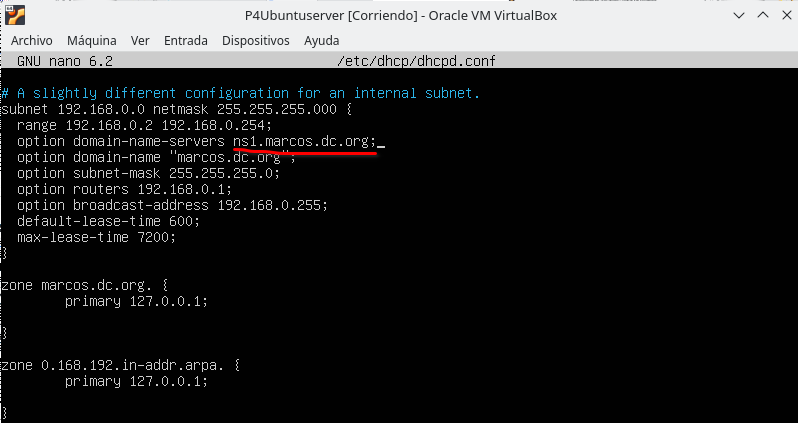


Imagen ilustrativa

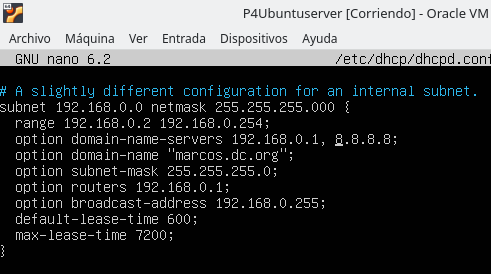


Imagen ilustrativa

Una vez aplicado esto, debemos hacer un “sudo service restart” para reiniciar el servicio dhcp y que se apliquen los cambios.



Imagen ilustrativa

Posteriormente hacemos “sudo nano /etc/sysctl.conf” y descomentamos la línea 28 para habilitar que google nos resuelva lo que nosotros no podremos:

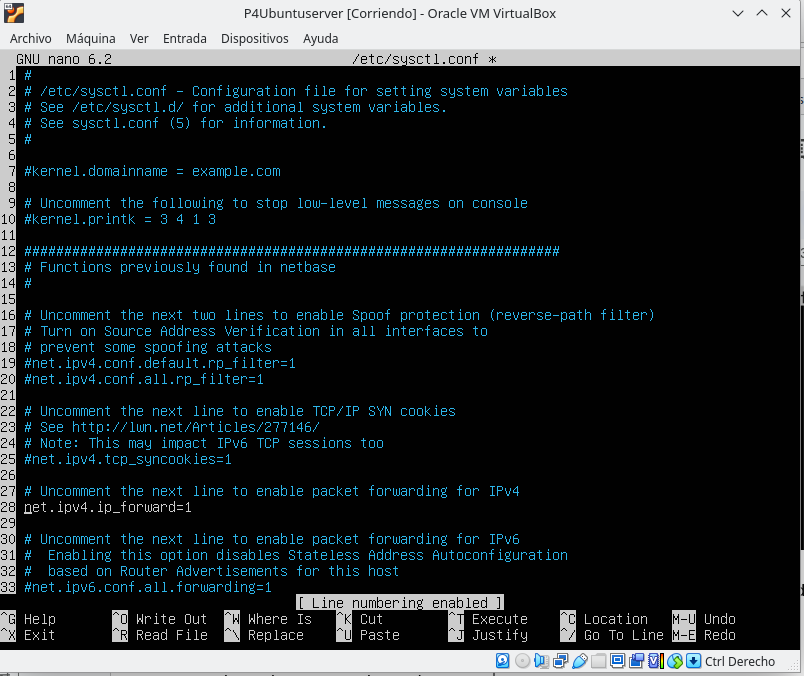


Imagen ilustrativa

Una vez modificado esto debemos hacer el comando “sudo sysctl -p” para que aplique los cambios realizados.



Imagen ilustrativa

El siguiente paso se puede hacer con “nftables” pero yo elegí hacerlo con el sistema de “iptables”.

Por lo que debemos crear un fichero en /usr/bin llamado “set\_iptables.sh” para configurar el enrutador mediante la herramienta iptables. También debemos hacer un chmod para que podamos modificarlo y ejecutarlo ya que luego entraremos en este archivo y crearemos la configuración:

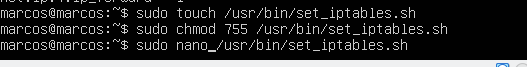


Imagen ilustrativa

Luego cómo se puede ver en la anterior captura hacemos un nano para meter las siguientes normas.

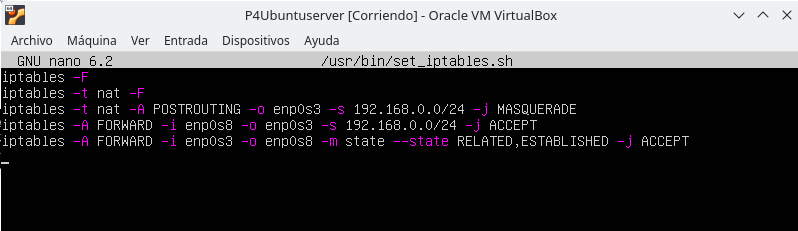


Imagen ilustrativa

A continuación dejo la misma captura pero con la explicación de cada línea del archivo para que asi quede mas claro los parámetros usados:

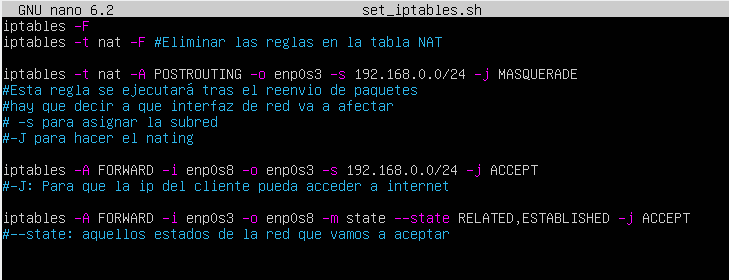


Imagen ilustrativa

Luego lanzamos este sh con el comando “sudo /usr/bin/set\_iptables.sh” y posteriormente hacemos el comando “sudo iptables -L” para comprobar que esta configuracion esta correcta:

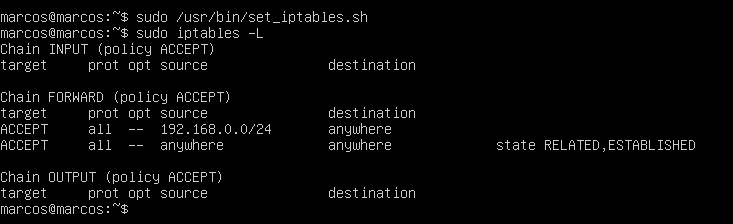


Imagen ilustrativa

## 

## - Utilizando un cliente con una sola interfaz en red interna:

## - Comprobar que se asigna la IP dentro de la subred suministrada

Para esto no metemos en el cliente en red interna y ponemos “ip a” y podemos comprobar que si se le asigna una ip:

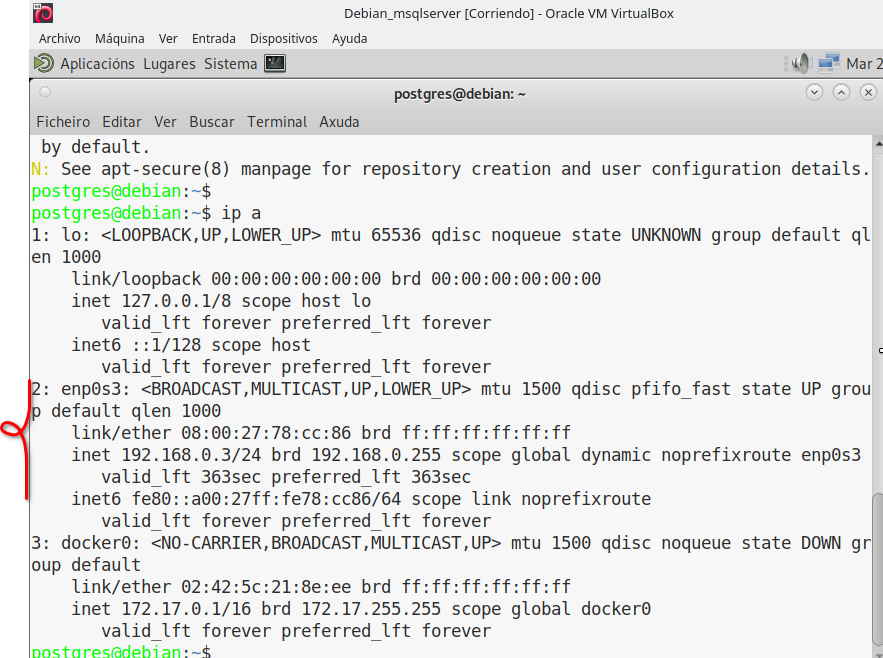


Imagen ilustrativa

## - Comprobar que se puede hacer ping a ips externas

Para realizar este paso primero hago dig a chatgpt.com para asi saber la ip y hacer un ping, como se puede ver a continuación:

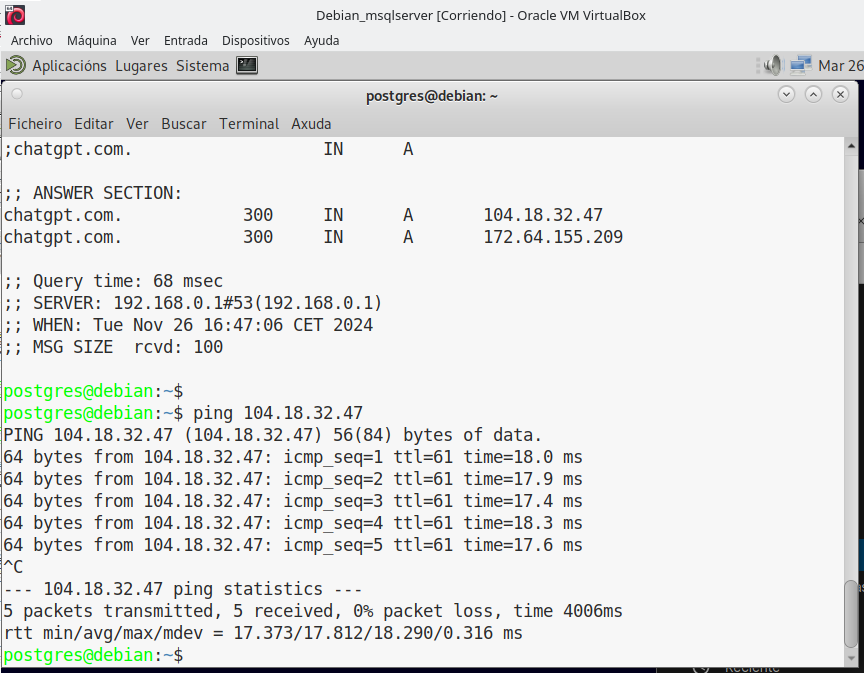


Imagen ilustrativa

## - Comprobar que se puede hacer ping y navegar a urls externas

Luego nos metemos a navegar por urls externas como la de esemtia:

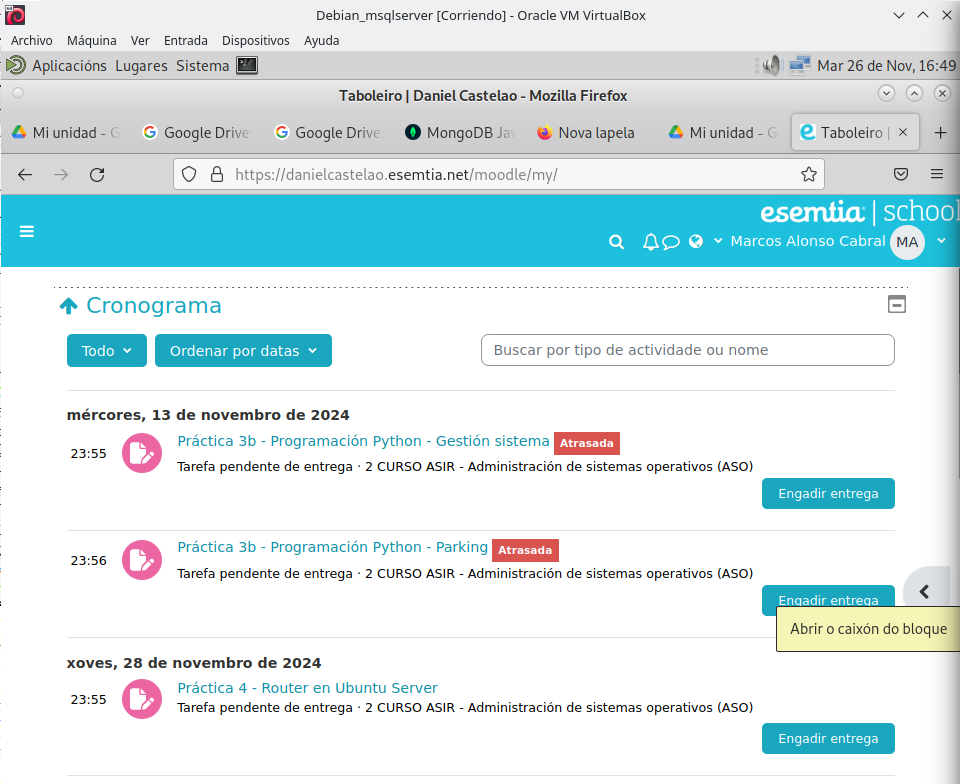


Imagen ilustrativa

Y podemos hacer ping a urls externas como por ejemplo chatgpt.com como vimos anteriormente:

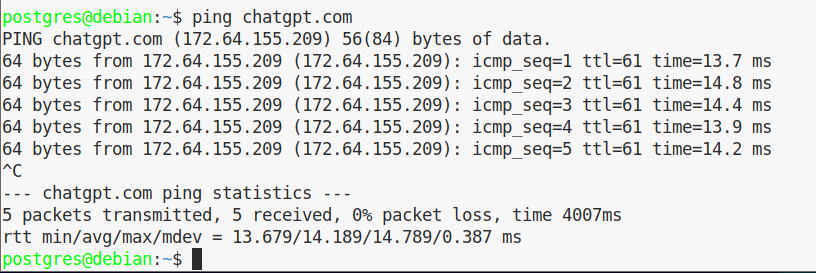


Imagen ilustrativa

## - Comprobar que se puede acceder a un servicio interno (Por ejemplo, un servidor web Apache)

Para instalar el servicio de apache2 en el servidor de la red interna debemos primero que todo hacer un “sudo apt update” y una vez actualizados los repositorios hacemos un “sudo apt install apache2” como podemos ver en la imagen:

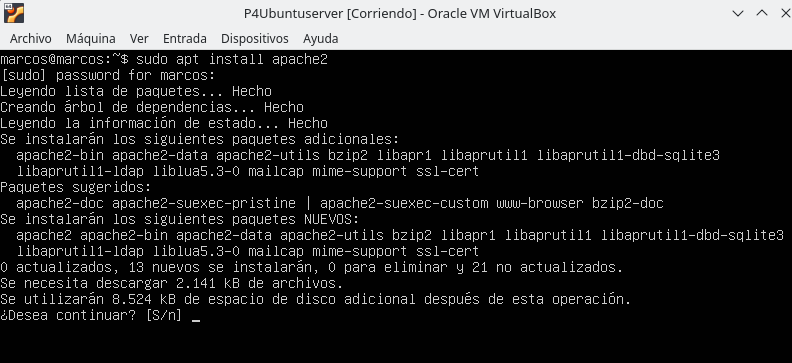


Imagen ilustrativa

Para verificar que el servidor que acabamos de instalar esta en ejecucion ponemos el comando “sudo systemctl status apache2”:

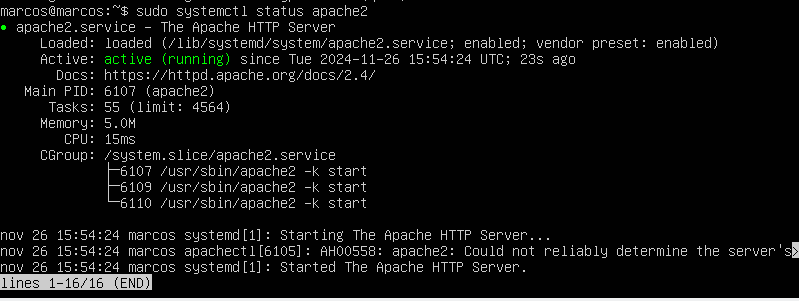


Imagen ilustrativa

Si no está activo podemos iniciarlo poniendo el comando “sudo systemctl start apache2”, aunque en mi caso ya se inició al instalarlo.

Y por último para que solo se pueda acceder mediante la red interna, ponemos el comando “sudo nano /etc/apache2/sites-available/000-default.conf” y en esta configuración de apache cambiamos la primera línea y en vez de un” \*” ponemos la “ip de la red del servidor” que en mi caso es la 192.168.0.0:

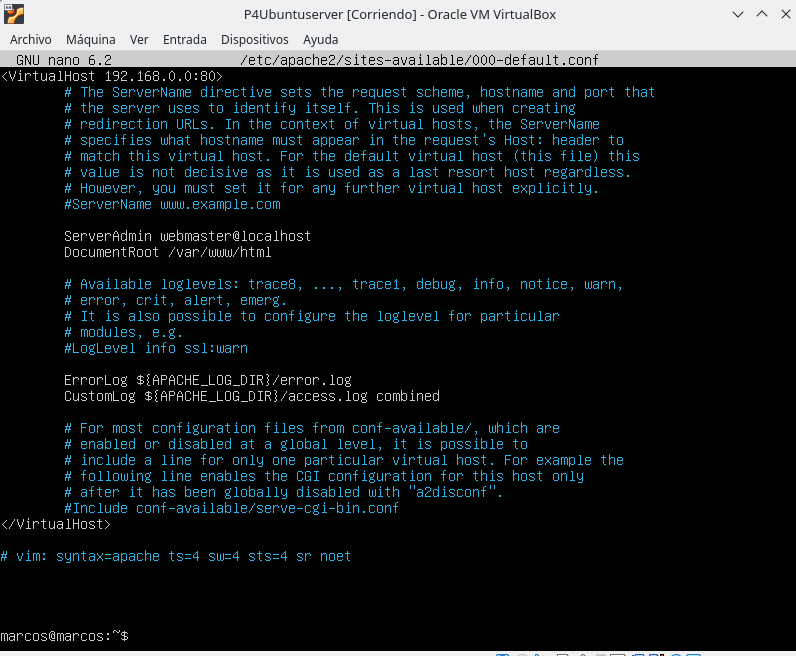


Imagen ilustrativa

Con esta configuración de apache, el servidor principal solo suministra este servicio a la red interna.

Una vez modificado y guardado este fichero, debemos poner el comando “sudo apache2ctl configtest” para comprobar que no tiene ningun error y luego hacemos un “sudo systemctl restart apache2” para reiniciar el servicio y que se aplique lo que acabamos de modificar:

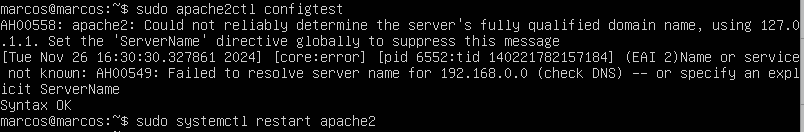


Imagen ilustrativa

Luego entramos en el cliente y ponemos la ip del servidor que es la 192.168.0.1 en mi caso. Y como podemos ver en esta última captura, el sistema muestra la página de apache que debería mostrar por defecto :

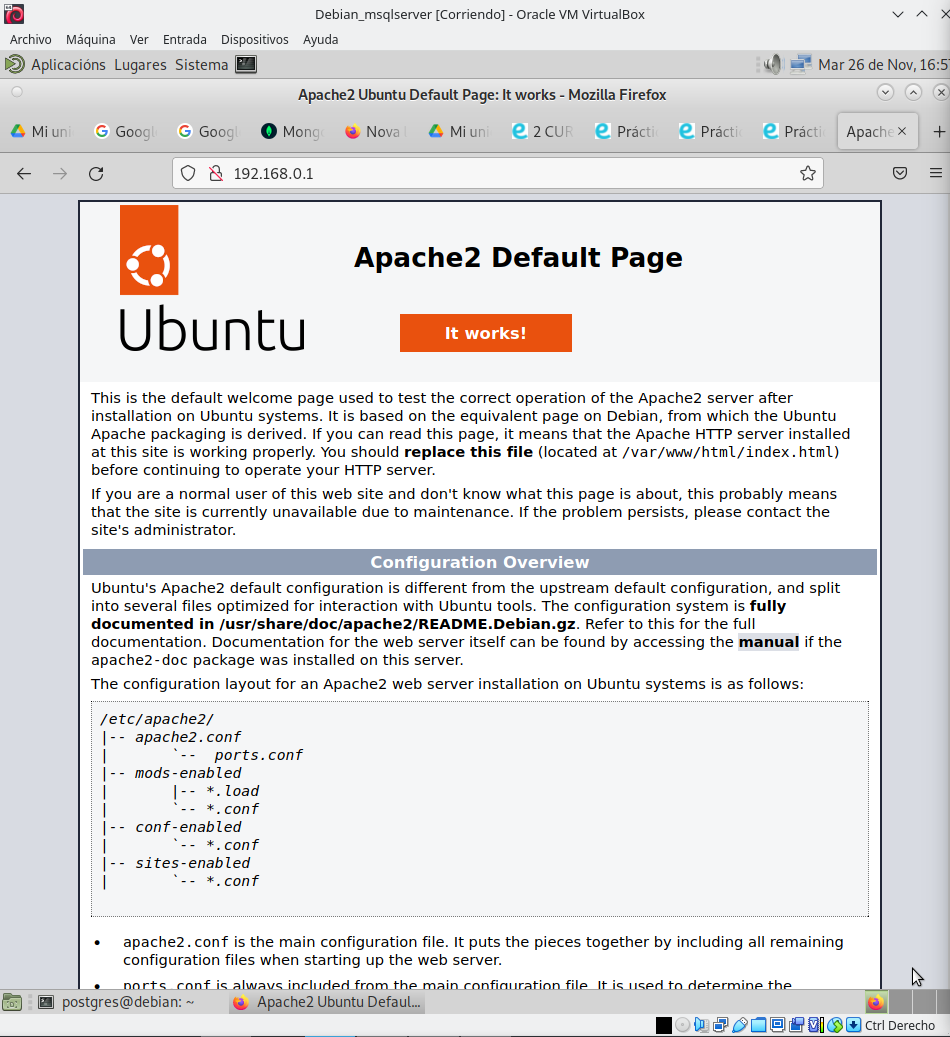


Imagen ilustrativa