

FACULTAD DE CIENCIAS BÁSICAS E INGENIERÍAS
SERVIDOR MAIL CON UBUNTU-SERVER Y WEBMIN
LINUX



JHOSSER FABIAN PEDRAZA CAMACHO
FERNEY JESÚS RODRIGUEZ FONSECA
TECNOLOGÍA EN SISTEMAS DE TELECOMUNICACIONES

UNIDADES TECNOLÓGICAS DE SANTANDER
BUCARAMANGA / SANTANDER
2017

INTRODUCCIÓN

El presente documento es una guía práctica para el desarrollo y ejecución de un servidor mail en Ubuntu Server 17.04 con apoyo en la plataforma gráfica Webmin, contará con un marco teórico breve para mayor comprensión de las condiciones en el montaje de dicho servidor, el proceso paso a paso con respaldo gráfico, descripción de los requisitos, dos anexos para desarrollar algunos de los requisitos anteriores mencionados si no son correspondidos por parte del lector, un ejemplo con parámetros en condiciones reales para resolver este servidor en un red de área local LAN y finalmente, la prueba y ejecución de dicho servidor mail para dos clientes en envío, lectura y respuesta de correos electrónicos de cada uno.

REFERENCIA TEÓRICA DEL SERVIDOR MAIL

Un servidor mail o de correo es una aplicación de red de computadoras ubicada en un servidor de internet, para prestar el servicio de correo electrónico. Este tiene tres componentes fundamentales:

- **Mail Transfer Agent (MTA)**

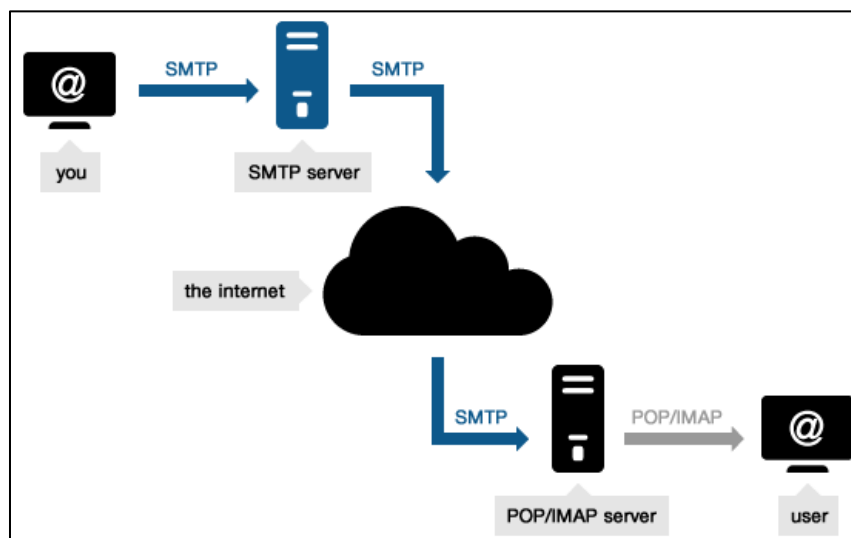
Este está encargado de transferir el correo de un ordenador a otro utilizando el protocolo SMTP, este tipo de agente solo permite el envío de dicho correo, lo más utilizados actualmente son el Postfix que poco complejo de configurar y fácil de instalar, SendMail este viene preinstalado en los sistemas LINUX pero son complejos de configurar ya que requieren basto conocimiento en dicho sistema, Qmail no es tan conocido y difícil de instalar, por último tenemos a Exim que viene instalado por defecto en los sistemas basados en Debian y también es muy eficiente. Todos estos MTA funcionan exactamente igual en cuanto a función aunque tiene ciertas diferencias en la arquitectura ya que fueron elaborados por diferentes autores.

- **Mail Delivery Agent (MDA)**

Está encargado de entregar el correo a los usuarios enviado por el MTA y gestionar la seguridad utilizando los protocolos IMAP y POP3, algunos ejemplos de estos servidores MDA son, Courier IMAP, Cyrus, Dovecot, Qpopper y Procmail.

- **Mail User Agent (MUA)**

Es el cliente de correo web o de escritorio encargado de recuperar los correos del servidor utilizando los protocolos SMTP, IMAP y POP3, es una interfaz que nos permite recuperarlos de los servidores MDA para que los tenga el usuario, algunos ejemplos son, Squirrelmail (web), Thunderbird (Ubuntu escritorio), Evolutions (Debian escritorio) y Outlook (Microsoft).



REQUISITOS PARA LA ELABORACIÓN DE UN SERVIDOR MAIL

La siguiente lista corresponde a los requisitos en estructura para poder elaborar un servidor mail (en área local) funcional.

0. Servicio de Webmin activo. (opcional)

Para iniciar se debe tener instalado y configurado el servicio de Webmin para facilitar algunos aspectos a la hora de configurar servidores en nuestro Ubuntu Server, esta información está descrita en el anexo 1.

1. Nombre de dominio resuelto.

En este apartado se empleará la elaboración de un servidor DNS con apoyo en Webmin para su posterior ejecución en el dominio del servidor mail, se utilizará como ejemplo `www.ubuntumail.ub` siendo el apartado de dominio para el correo “usuario”@ubuntumail.ub, esta información está descrita en el anexo 2.

2. Sitio web funcional con soporte de PHP. (opcional)

Este requisito es opcional si se trata de desarrollar solo un servicio a entono de escritorio y no de Web. En todo caso se recomienda el empleo del servidor Apache2 para su elaboración, esta información está descrita en el anexo 3.

3. Servidor de correo de salida SMTP. (MTA)

Se emplea el servidor Postfix para este caso.

4. Servidor de entrega de correos IMAP POP3. (MDA)

Se empleará el servidor Dovecot para este caso.

5. Clientes de correo de tipo escritorio que soporten SMTP, IMAP y POP3. (MUA)

Se empleará el programa Thunderbird para este caso.

DETALLES DEL EJEMPLO.

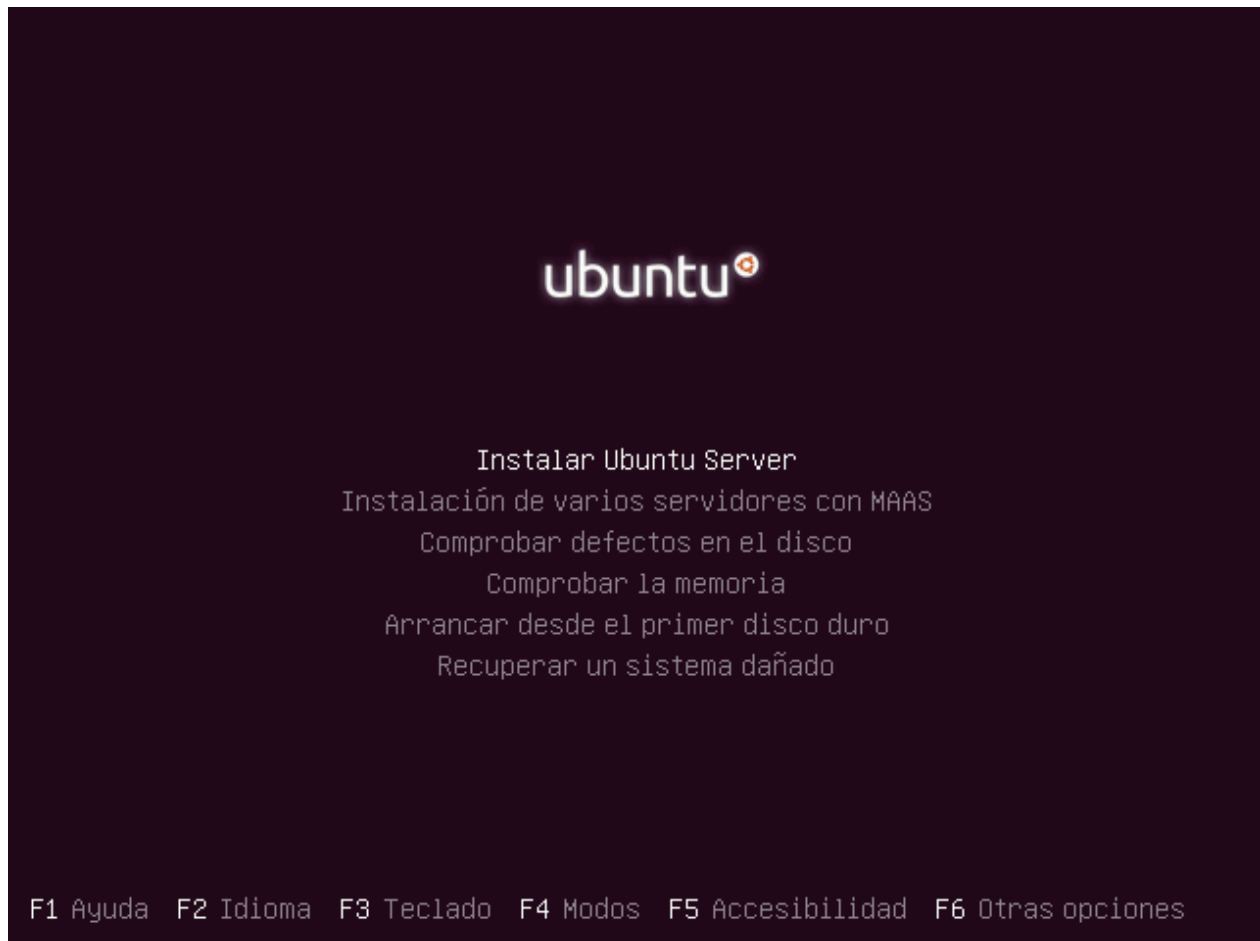
El servidor mail contará con la dirección IP estática 192.168.0.18/24, Gateway 192.168.0.1, servidor de DNS (con la misma IP), nombre de dominio `www.ubuntumail.ub` (también para protocolo FTP, POP3, IMAP y SMTP) servicios MTA, MDA Y MUA acondicionados con los servidores Postfix, Dovecot y Thunderbird respectivamente. El user de instalación del servidor se llamará “mailserver” y la password será `ubuntuserverJP`.

El servidor tendrá dos clientes “user_1” y user_2” ambos con password 1234, cabe aclarar que el usuario de instalación también puede contar con servicio mail así como cualquier otro usuario asociado a la máquina.

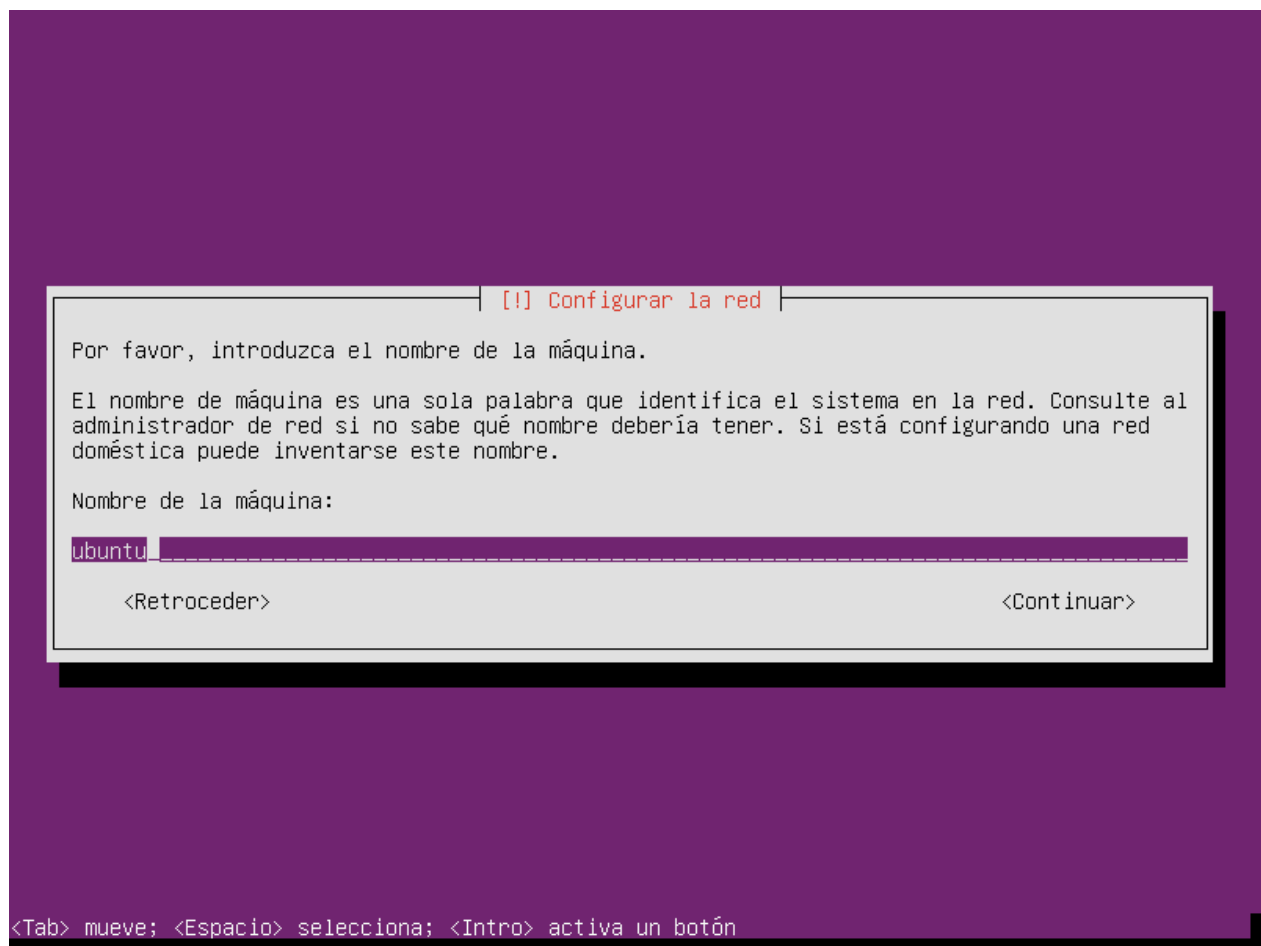
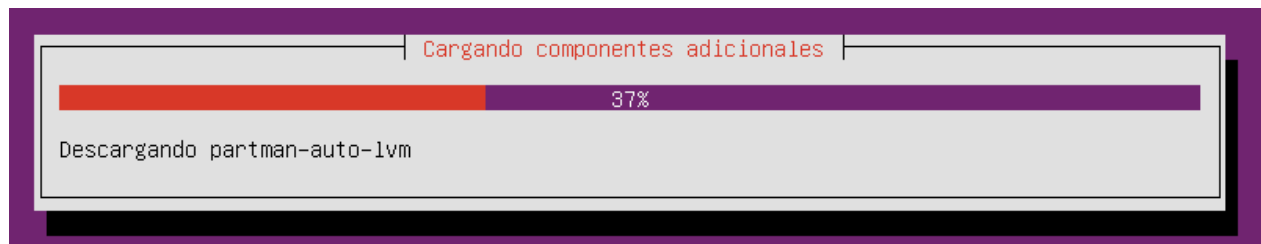
SERVIDOR MAIL UBUNTU SERVER 17.04

Instalación y configuración de Ubuntu Server 17.04

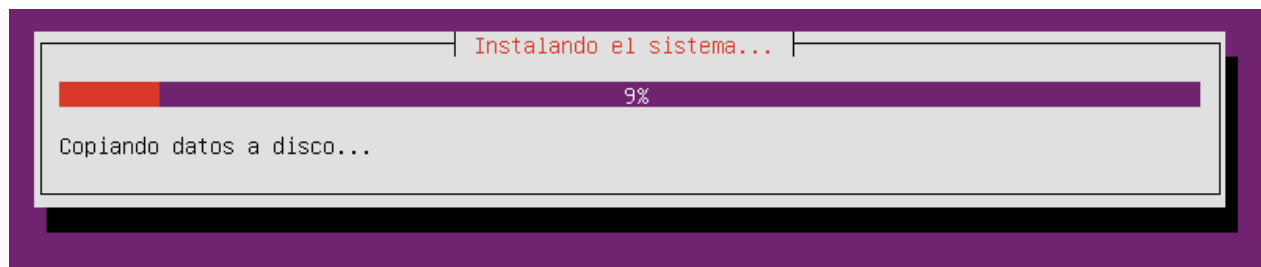
Para empezar necesitaremos la correcta instalación y configuración inicial del S.O Ubuntu Server 17.04. Primero arrancaremos un ordenador con la imagen ISO montada en la unidad de CD o USB boteable que corresponda a la capacidad del tamaño del archivo, o bien una máquina virtual donde se esté ejecutando la imagen ISO desde la ruta donde esta esté guardada como en este caso.



Daremos enter a esta opción “Instalar Ubuntu Server”, luego algunos módulos empezarán a cargarse como lo muestra la siguiente figura.

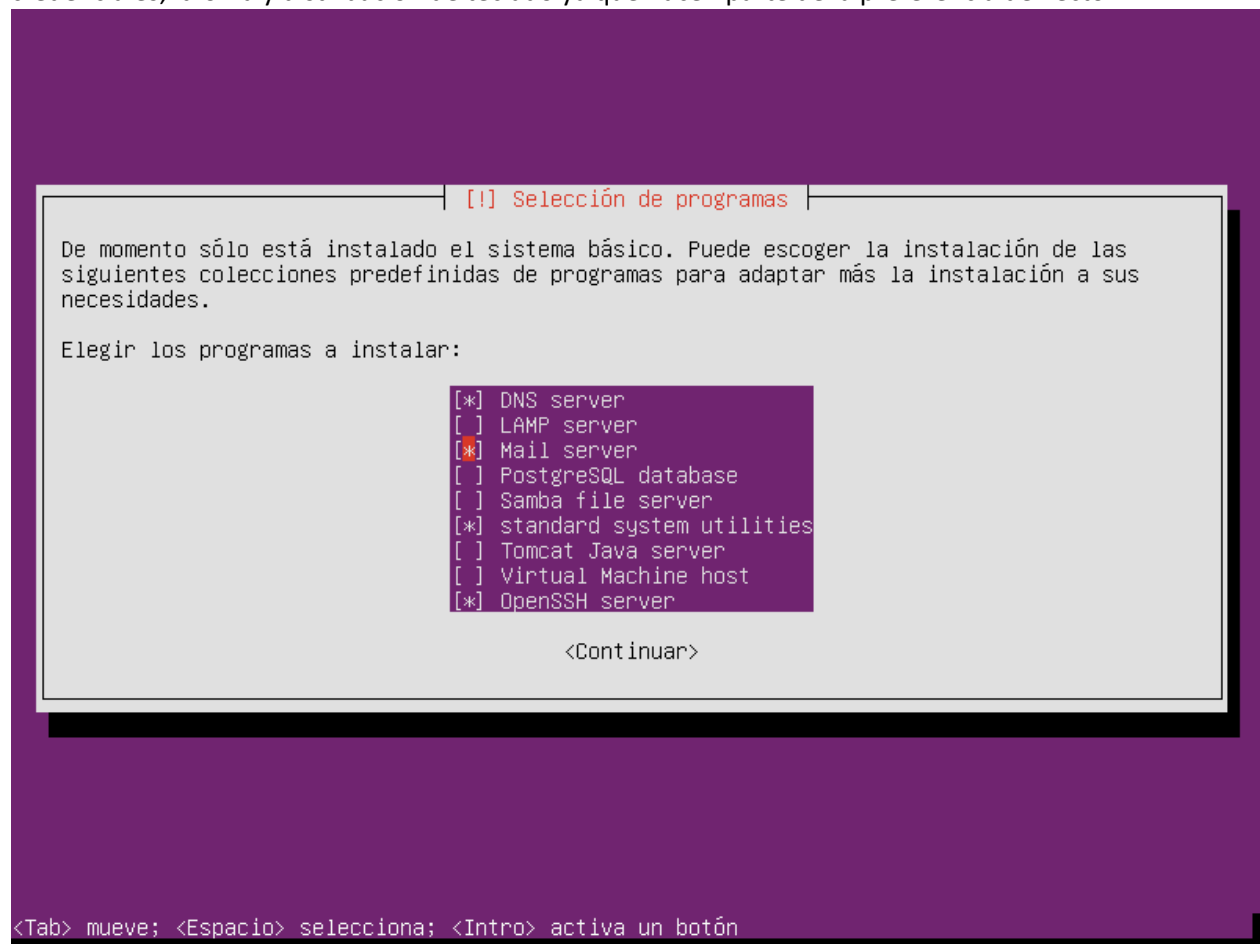


En este apartado el asistente de instalación nos solicita el nombre de la máquina para la configuración de la red, “ubuntu” recuerde considerar este nombre para futuras configuraciones en otros servicios.



Se seguirán cargando más módulos de instalación y configuración, luego de unos minutos llegaremos al apartado como lo muestra la siguiente figura.

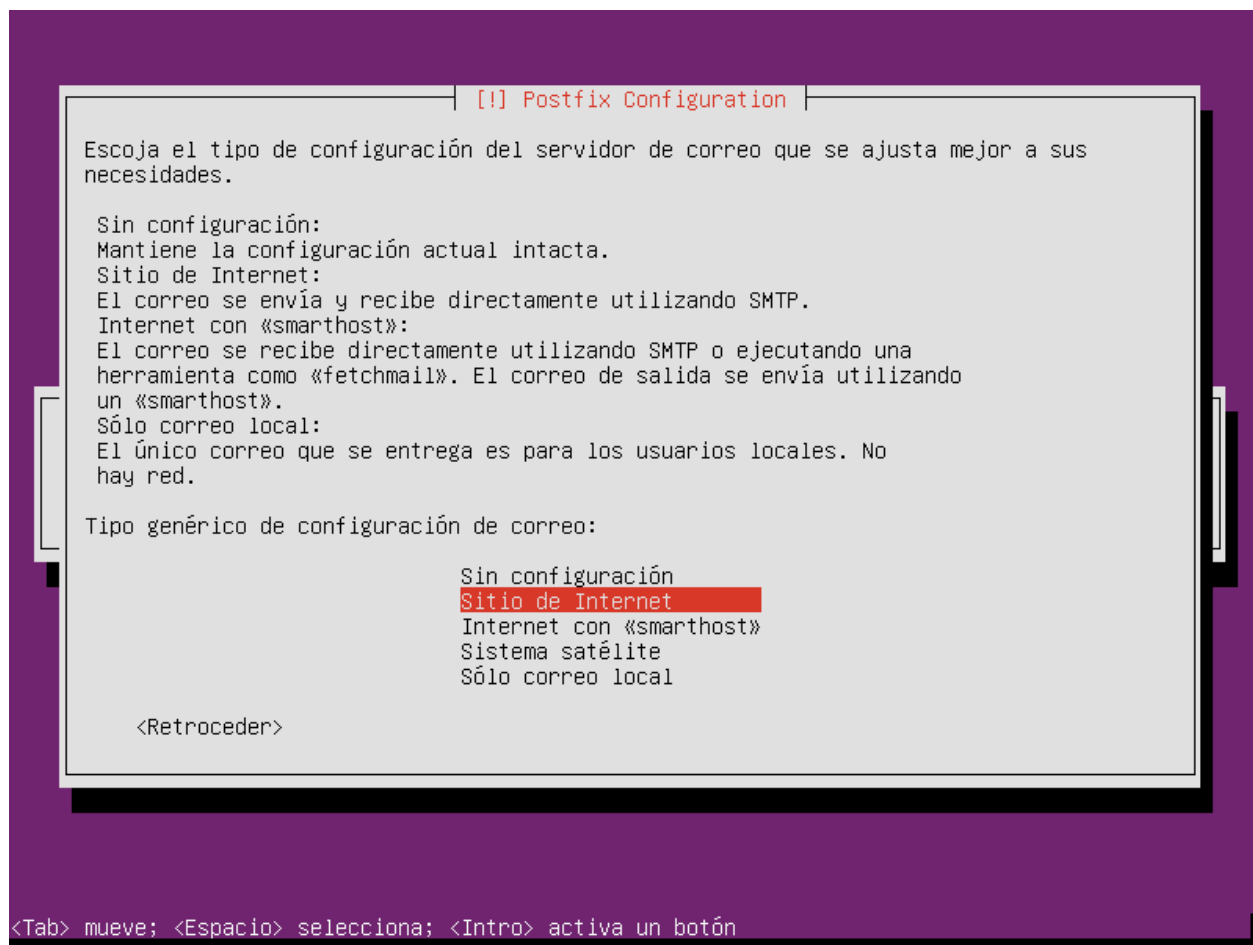
Cabe aclarar que obviaremos la configuración de nombre de usuario de instalación y otros parámetros de credenciales, idioma y distribución de teclado ya que hacen parte de la preferencia del lector.



En este módulo encontraremos una lista de los diferentes programas y servicios que podremos instalar durante la ejecución de la instalación del S.O directamente, este apartado es equivalente a ejecutar en el terminal el comando **tasksel** donde se encontraran estos y otros servicios más. Para este ejemplo crearemos estas condiciones de entrada seleccionando los servidores y programas que nos interesa como lo son el DNS server y el Open SSH server (el anexo 0.1 describe la importancia de la utilización de una conexión al servidor en un cliente dentro de la red por medio de SSH), seleccionamos los ítems con barra espaciadora e intro o enter para instalarlos una vez los hayamos seleccionado.

Configuración del servicio Postfix MTA.

Cabe resaltar que esta configuración aparece en medio de la instalación del S.O Ubuntu Server 17.04 ya que así se decidió en el desarrollo de este ejemplo, pero no es indispensable que así se realice si desea reconfigurar los parámetros del servicio Postfix una vez ya instalado el S.O dentro del termina puede ejecutar el siguiente comando **sudo dpkg-reconfigure postfix** y obtendrá el mismo módulo que se muestra en la siguiente imagen.



Una vez aquí se procede a configurar el Postfix (MTA) seleccionamos la opción **Sitio de Internet** y presionamos enter. Posterior a esto nos pedirá un nombre de dominio para relver el servidor Postfix en este caso tenemos el dominio **ubuntumail.ub**, se digita y se sigue la instalación (los apartados que no se mencionan aquí simplemente se dejan en blanco, es decir se omiten y se presiona enter)

```
[ OK ] Started Auto import assertions from block devices.
[ OK ] Started System Logging Service.
      Starting Authorization Manager...
[ OK ] Started Authorization Manager.
[ OK ] Started Accounts Service.
[ OK ] Started Raise network interfaces.
[ OK ] Reached target Network.
      Starting Network Name Resolution...
[ OK ] Started BIND Domain Name Server.
[ OK ] Reached target Host and Network Name Lookups.
      Starting Dovecot IMAP/POP3 email server...
```

Una vez ejecutada esta acción se empezarán a mostrar la carga y el estado de los servicios que requieren el servidor Mail como lo son el MTA y MDA, es decir automáticamente al momento de instalar el servidor mail dentro del apartado de Tasksel o el apartado de instalación dentro de la configuración de programas y servicios que se desean, se procesan los servicios de Postfix (visto anteriormente) y Dovecot MTA y MDA respectivamente, los cuales son los que los sistemas basados en Debian prefieren y traen por defecto.


```

Ubuntu 17.04 ubuntu tty1

ubuntu login: mailserver
Password:
Welcome to Ubuntu 17.04 (GNU/Linux 4.10.0-19-generic i686)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

 * Ubuntu 12.04 LTS ('precise') end-of-life was April 28, 2017
   ongoing security updates for 12.04 are available with Ubuntu Advantage
   http://j.mp/U1204esm

Pueden actualizarse 45 paquetes.
32 actualizaciones son de seguridad.

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

mailserver@ubuntu:~$ _

```

Con esto termina el proceso de instalación del S.O Ubuntu Server 17.04 y el servidor Mail, simplemente al iniciarse logeamos con nuestro usuario “mailserver” y la password.

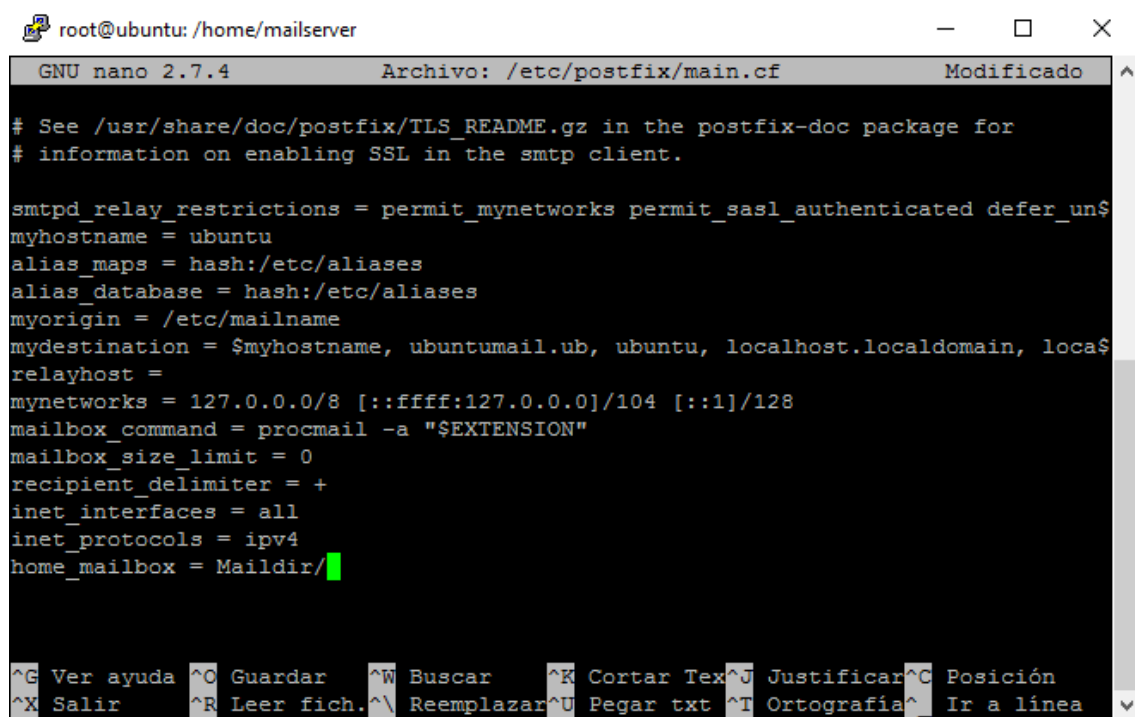
Una vez llegados a este punto tendremos nuestro S.O y nuestros servicios MTA y MDA pero no configurados por completos (es decir la configuración del Postfix y Dovecot)

Antes que nada nuestro servidor debe tener conexión con los ordenadores clientes y preferiblemente conexión a internet mediante un router, para una máquina virtual esta condición ser resolverá en el anexo 0 al final de este documento, también es preciso aclarar que utilizaremos el servidor gráfico para sistemas Debian Webmin cuya instalación y configuración está desarrollada en el anexo 1, luego de la ejecución de estas dos condiciones pasaremos a instalar un servidor DNS (en caso de que el lector no lo tenga), la configuración e instalación de este servidor se ven resueltas en el anexo 2, también un servidor web con soporte PHP Apache2 (opcional) resuelto en el anexo 3. Todos estos documentos se encuentran al final de este manual ya que nuestro objetivo principal es concentrarnos en el servidor mail y sus parámetros, pero para esto tendremos que resolver primero las condiciones planteadas al principio de este documento.

El acceso al servidor será mediante SSH desde un ordenador conectado a la misma red LAN (véase anexo 0.1).

Aclarado esto procederemos complementar la configuración del MTA y MDA Postfix y Dovecot.

Entraremos al archivo de configuración de postfix mediante el editor “nano” con el comando (una vez en modo root) **nano /etc/postfix/main.cf**, La siguiente figura muestra la estructura del archivo.



```
GNU nano 2.7.4 Archivo: /etc/postfix/main.cf Modificado
# See /usr/share/doc/postfix/TLS_README.gz in the postfix-doc package for
# information on enabling SSL in the smtp client.

smtpd_relay_restrictions = permit_mynetworks permit_sasl_authenticated defer_un$
myhostname = ubuntu
alias_maps = hash:/etc/aliases
alias_database = hash:/etc/aliases
myorigin = /etc/mailname
mydestination = $myhostname, ubuntu.ub, ubuntu, localhost.localdomain, loca$
relayhost =
mynetworks = 127.0.0.0/8 [::ffff:127.0.0.0]/104 [::1]/128
mailbox_command = procmail -a "$EXTENSION"
mailbox_size_limit = 0
recipient_delimiter = +
inet_interfaces = all
inet_protocols = ipv4
home_mailbox = Maildir/
```

Aquí agregaremos nuestro dominio (si no lo está) como se ve en la línea de texto número ocho **mydestination = ubuntu.ub** respetaremos la sintaxis de la línea separando cada dominio con comas y un espacios, pondremos en **inet_protocols = ipv4**. Luego al final del documento agregaremos esta línea **home_mailbox = Maildir/** que será la carpeta que contendrá los correos si el servicio es MTA por protocolo IMAP, es decir, el usuario no guarda el correo si no el servidor. Guardamos los cambios [Ctrl X] [s] y enter.

Ahora configuraremos el MDA Dovecot POP3 (protocolo de entrega de correo), editamos con nano el archivo **nano /etc/dovecot/dovecot.conf** y agregamos la última línea **disable_plaintext_auth = no** y guardamos nuevamente los cambios.



```
GNU nano 2.7.4 Archivo: /etc/dovecot/dovecot.conf Modificado
#quota = mysql:/etc/dovecot/dovecot-dict-sql.conf.ext
#expire = sqlite:/etc/dovecot/dovecot-dict-sql.conf.ext
)
# Most of the actual configuration gets included below. The filenames are
# first sorted by their ASCII value and parsed in that order. The 00-prefixes
# in filenames are intended to make it easier to understand the ordering.
!include conf.d/*.conf
# A config file can also tried to be included without giving an error if
# it's not found:
!include_try local.conf
disable_plaintext_auth = no
```

[Ctrl X] [s] y enter.

Para asegurarnos del funcionamiento reiniciaremos ambos servicios como lo muestra la siguiente imagen.

```
root@ubuntu: /home/mailserver

root@ubuntu:/home/mailserver# service dovecot restart
root@ubuntu:/home/mailserver# service dovecot status
• dovecot.service - Dovecot IMAP/POP3 email server
  Loaded: loaded (/lib/systemd/system/dovecot.service; enabled; vendor preset:
  Active: active (running) since Tue 2017-05-16 19:56:39 -05; 5s ago
  Docs: man:dovecot(1)
        http://wiki2.dovecot.org/
  Process: 2890 ExecStop=/usr/bin/doveadm stop (code=exited, status=0/SUCCESS)
  Process: 2895 ExecStart=/usr/sbin/dovecot (code=exited, status=0/SUCCESS)
  Main PID: 2900 (dovecot)
  Tasks: 4 (limit: 4915)
  Memory: 1.9M
  CPU: 172ms
  CGroup: /system.slice/dovecot.service
          └─2900 /usr/sbin/dovecot
            └─2901 dovecot/anvil
              └─2902 dovecot/log
                └─2905 dovecot/config

may 16 19:56:39 ubuntu systemd[1]: Starting Dovecot IMAP/POP3 email server...
may 16 19:56:39 ubuntu systemd[1]: dovecot.service: PID file /var/run/dovecot/ma
may 16 19:56:39 ubuntu dovecot[2900]: master: Dovecot v2.2.27 (c0f36b0) starting
may 16 19:56:39 ubuntu systemd[1]: Started Dovecot IMAP/POP3 email server.
lines 1-21/21 (END)
```

Bajo el comando **service dovecot restart/status** y **service postfix restart/status** podemos reiniciar los servicios para que se carguen los módulos y configuraciones.

Ahora, según la descripción de los detalles del ejemplo debemos configurar la red de nuestro servidor con una IP estática, primero el router en nuestra LAN nos dará una IP automática por DHCP tomaremos esta y la definiremos estática para evitar problemas de reconocimiento en próximas conexiones del servidor a la LAN. Existen muchas formas de cambiar los parámetros de las interfaces de red, podremos visualizarlas bajo el comando de **ifconfig (nombre de interfaz)** o bien solo **ifconfig**, veremos los parámetros de red de cada interfaz, para configurarlo con los parámetros que se desean entraremos con el editor nano a la siguiente ruta de archivo **nano /etc/network/interfaces**.

```
GNU nano 2.7.4          Archivo: /etc/network/interfaces

# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

source /etc/network/interfaces.d/*

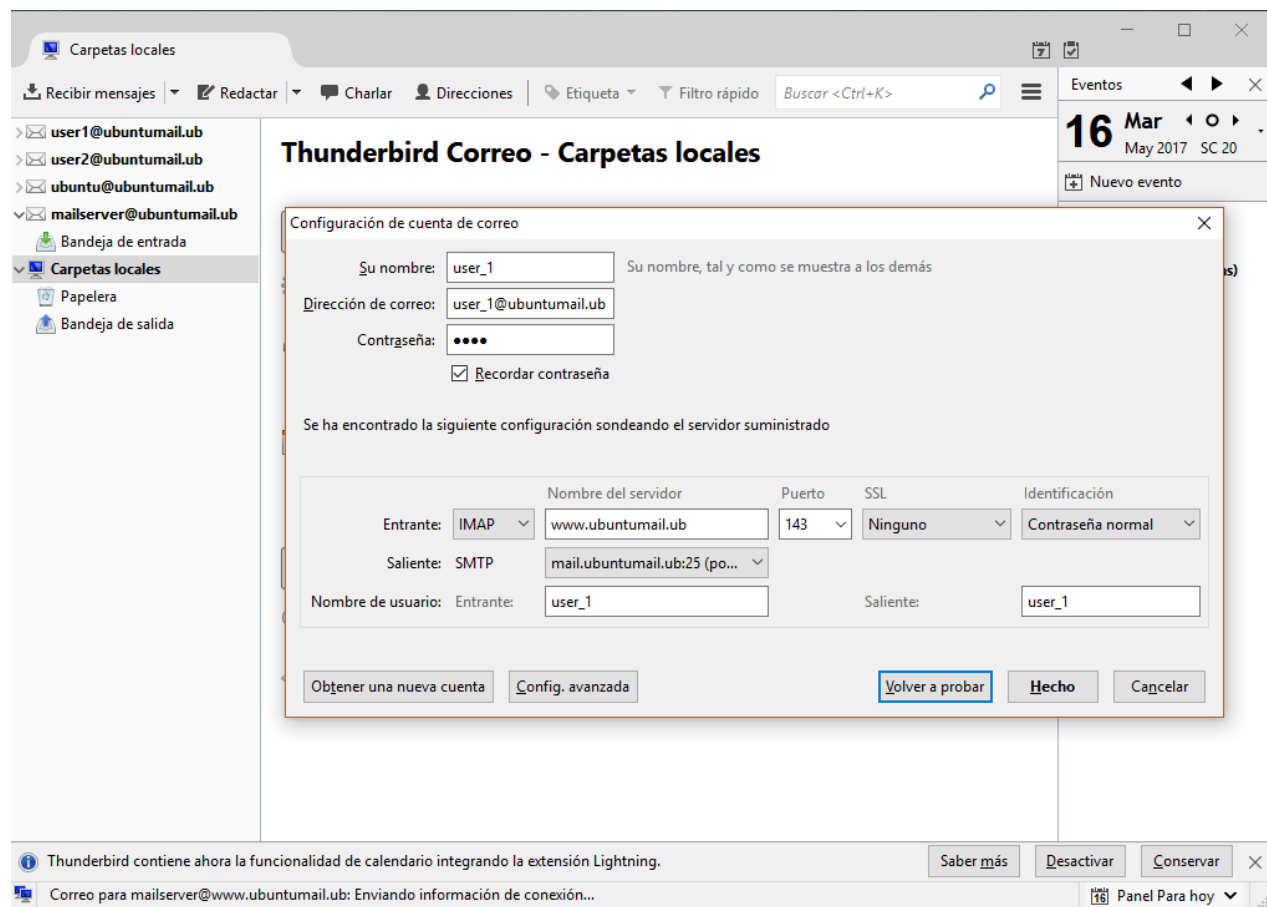
# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
iface enp0s3 inet static
    address 192.168.0.18
    netmask 255.255.255.0
    network 192.168.0.0
    gateway 192.168.0.1
    dns-nameservers 192.168.0.18
    dns-domain ubuntuub
```

Los parámetros establecidos aquí pertenecen a la interfaz enp0s3 su inet es estática y ya tiene resuelto los sistemas de DNS.

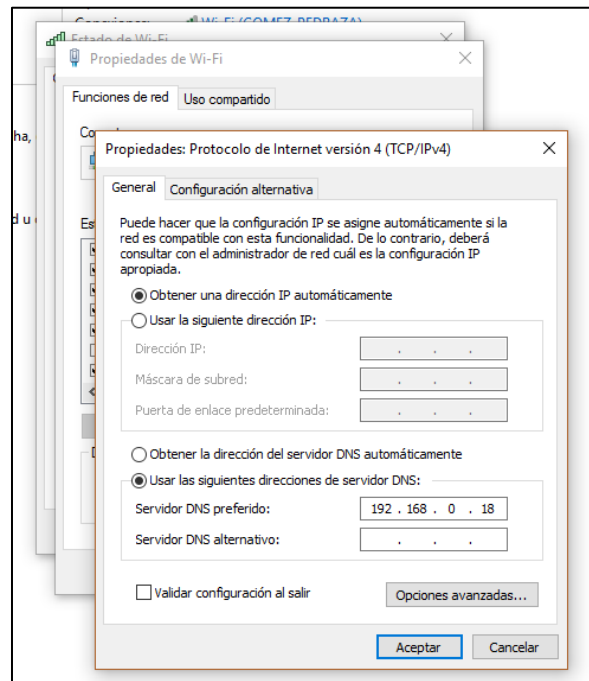
Bien, ahora procederemos a configurar los clientes del servidor mail, estos son creados en el servidor, es decir cada usuario del servidor tendrá ahora su propio correo como en este ejemplo el usuario de instalación mailserver@ubuntumail.ub o el usuario root como root@ubuntumail.ub. Para este ejemplo crearemos dos usuarios, user_1 y user_2 mediante la condición de creación de usuarios que prefiera, nosotros usaremos el servicio de Webmin para esto o bien podremos hacerlo por medio de **useradd** y se configuran los parámetros descritos en la sintaxis de dicho comando.

Ahora configuraremos los servicios de MUA de Thunderbird con los dos clientes y el servidor. Agregaremos una dirección de correo existente y la configuraremos con los parámetros que se muestran en la siguiente imagen.

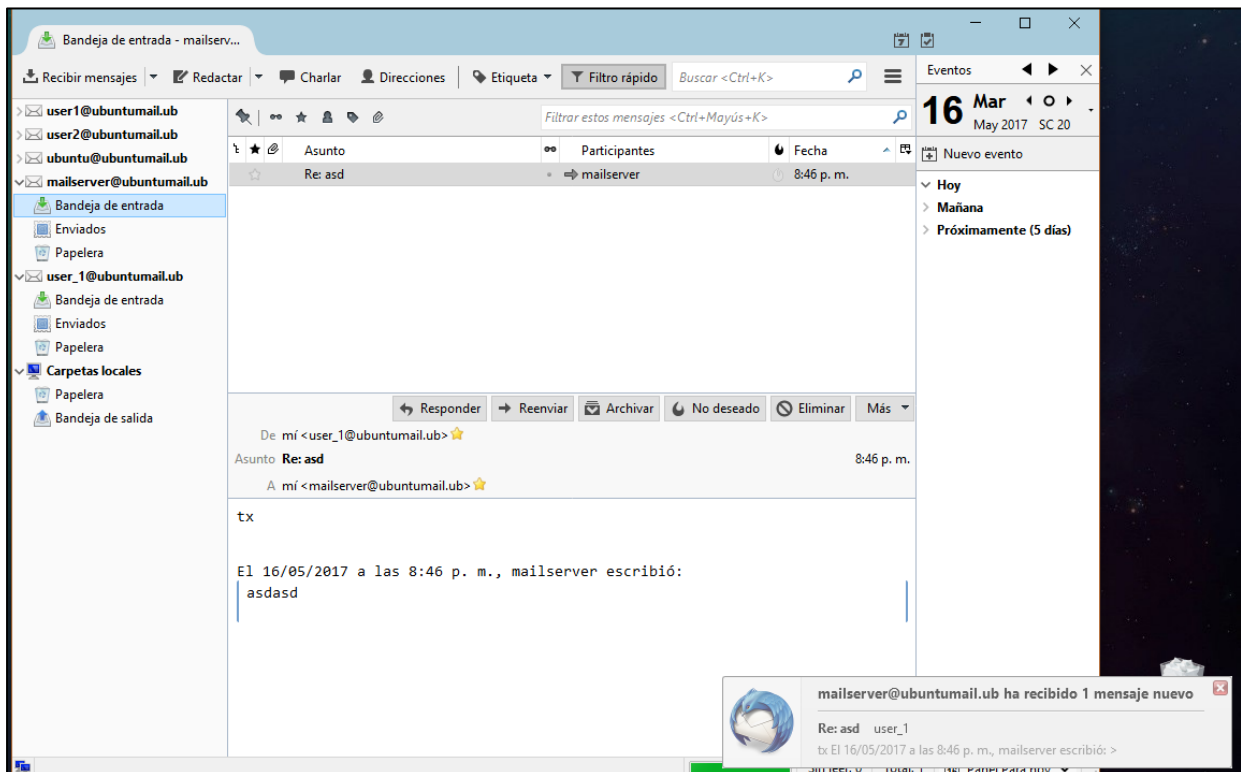


Estaremos configurando la cuenta para el user_1, su dirección mail pasaría a ser user_1@ubuntumail.ub y la contraseña la que se definió en la creación del usuario, cuando se prueba la conexión en la creación de la cuenta en Thunderbird se pide agregar una excepción de seguridad, podremos configurar por protocolo IMAP o POP3 ya que todo lo está resolviendo el dominio ubuntuuser.ub que tiene soporte para todos estos protocolos.

Los PC clientes deben estar conectados a la misma LAN del servidor y su DNS definido con la IP del servidor.



Ya preparado el cliente probaremos el servidor mail mediante Thunderbird con dos cuentas cualquiera creada en el servidor con anterioridad.



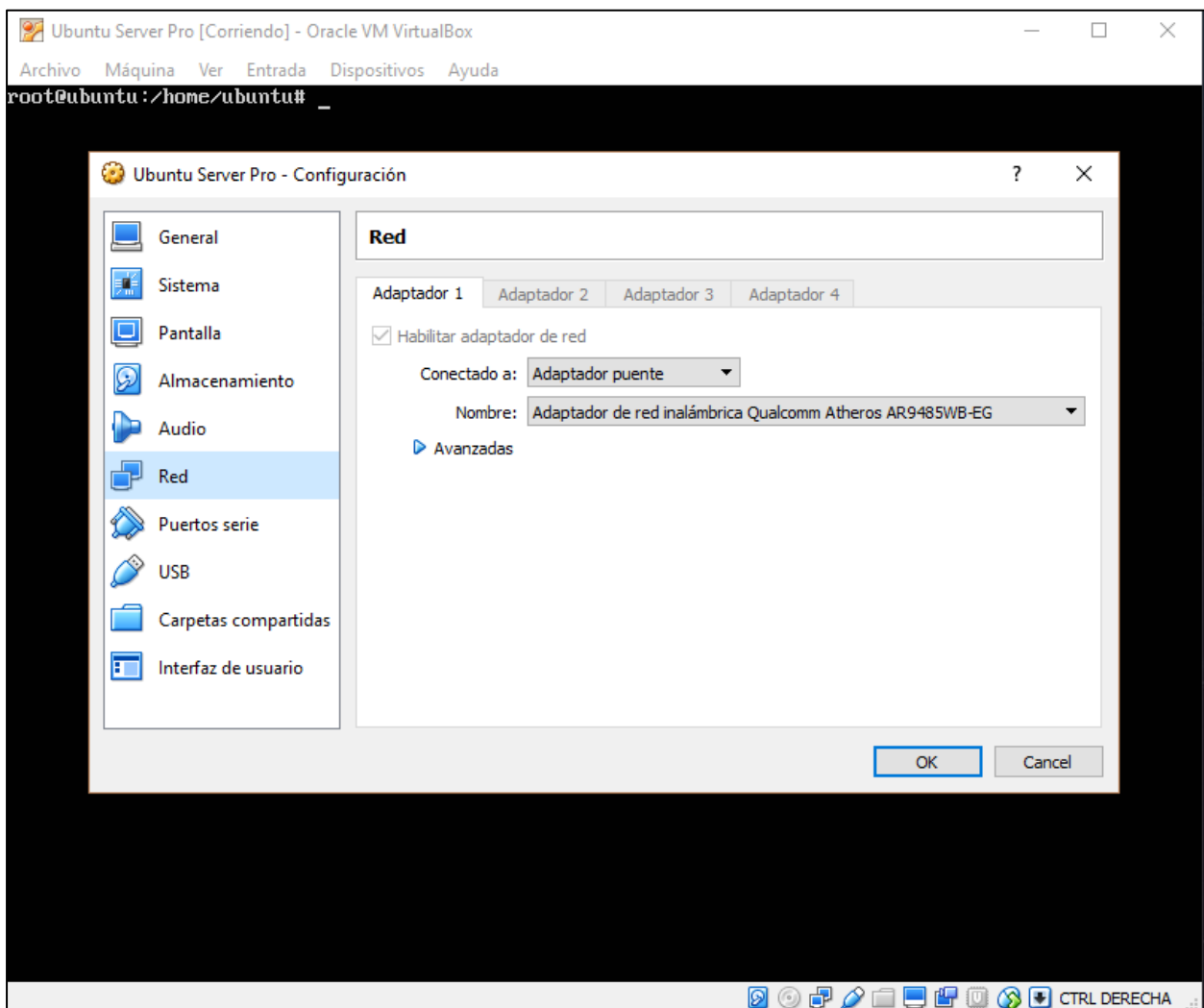
Esto define el correcto funcionamiento del servidor mail entre las cuentas mailserver@ubuntumail.ub y user_1@ubuntumail.ub.

[Anexo 0]

CONFIGURACIÓN DE RED EN MÁQUINA VIRTUAL

En este apartado realizaremos la configuración para que la máquina virtual (en nuestro caso) tenga conexión a la red de área local en la que queramos ejecutar el servidor, también el modo de acceso a dicho servidor por medio de SSH, como se vio anteriormente en el modelo de instalación del S.O Ubuntu server 17.04, instalamos el servicio de SSH.

Inicialmente las máquinas virtuales tienen conexión a internet y conexión unidireccional con otros dispositivos en la LAN, es decir podemos tener acceso a los host de la red pero desde los host no lo tendremos a la máquina virtual, para solucionar dicho inconveniente tendremos que efectuar la siguiente configuración.



Vamos a la configuración de red del Virtual Box y cambiamos la conexión **NAT** a **Adaptador puente** y seleccionamos el adaptador al que tengamos acceso a internet o acceso a la red.

```
root@ubuntu:/home/ubuntu# service networking restart
```

Una vez configurado esto ejecutaremos el comando **service networking restart** para reiniciar los servicios.

```

root@ubuntu:/home/ubuntu# ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.2.106 netmask 255.255.255.0 broadcast 192.168.2.255
    inet6 fe80::a00:27ff:fe12:e7a6 prefixlen 64 scopeid 0x20<link>
    ether 08:00:27:12:e7:a6 txqueuelen 1000 (Ethernet)
    RX packets 537 bytes 100957 (100.9 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 338 bytes 30864 (30.8 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Bucle local)
    RX packets 457 bytes 37105 (37.1 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 457 bytes 37105 (37.1 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

root@ubuntu:/home/ubuntu#

```

Miraremos el estado de la interfaz de red y veremos que se han cargado los nuevos módulos asignados por el router de esta, es recomendable que los servidores tengan direcciones estáticas así que se configurarán de este modo como lo evidencia la siguiente imagen editando la estructura del archivo **interfaces** mediante **nano /etc/network/interfaces**.

```

GNU nano 2.7.4 Archivo: /etc/network/interfaces

# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

source /etc/network/interfaces.d/*

# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
auto enp0s3
iface enp0s3 inet static
    address 192.168.2.106
    netmask 255.255.255.0
    network 192.160.2.0
    gateway 192.168.2.100

[ 16 líneas leídas ]
Ver ayuda  Guardar  Buscar  Cortar Text  Justificar  Posición  Pág. ant.
Salir  Leer fich.  Reemplazar  Pegar txt  Ortografía  Ir a línea  Pág. sig.

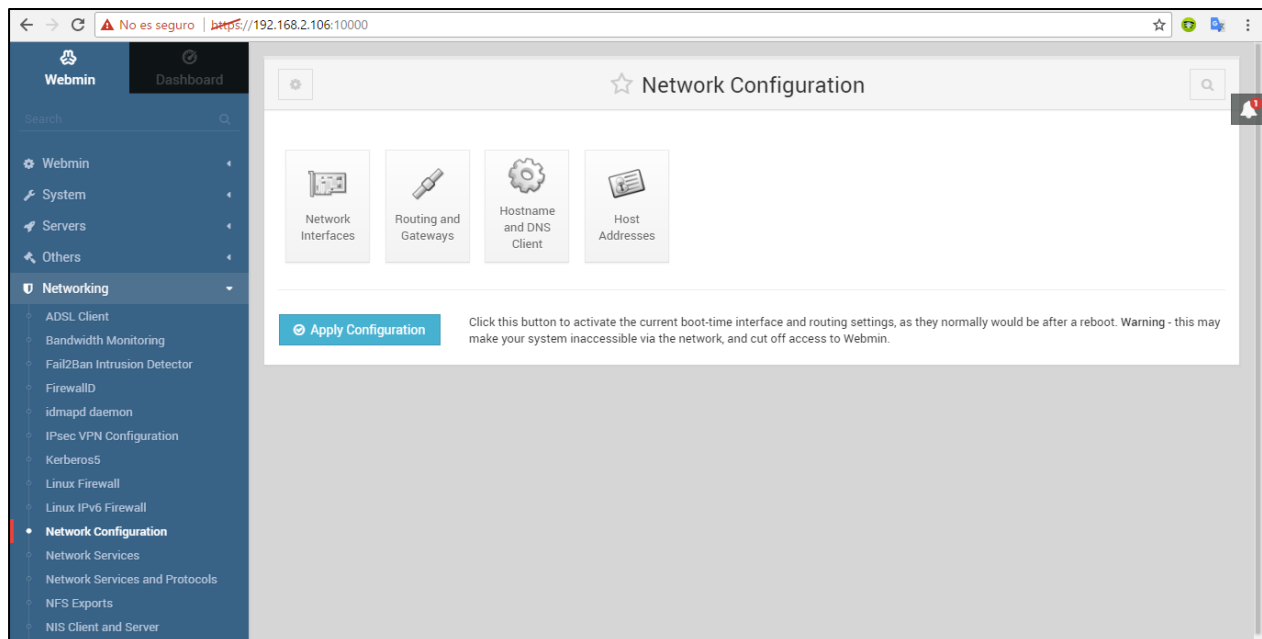
```

Una vez configurada la red estática y nuevamente reiniciado el servicio de Networking, haremos un “ping” a una página web cualquiera, si esto resulta, quiere decir que todos los parámetros de conexión están bien.

```
root@ubuntu:/home/ubuntu# ping google.com
PING google.com (216.58.222.206) 56(84) bytes of data.
64 bytes from bog02s05-in-f14.1e100.net (216.58.222.206): icmp_seq=1 ttl=250 time=25.1 ms
64 bytes from bog02s05-in-f14.1e100.net (216.58.222.206): icmp_seq=2 ttl=250 time=19.4 ms
64 bytes from bog02s05-in-f14.1e100.net (216.58.222.206): icmp_seq=3 ttl=250 time=23.0 ms

--- google.com ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2006ms
rtt min/avg/max/mdev = 19.446/22.554/25.135/2.355 ms
^Croot@ubuntu:/home/ubuntu#
```

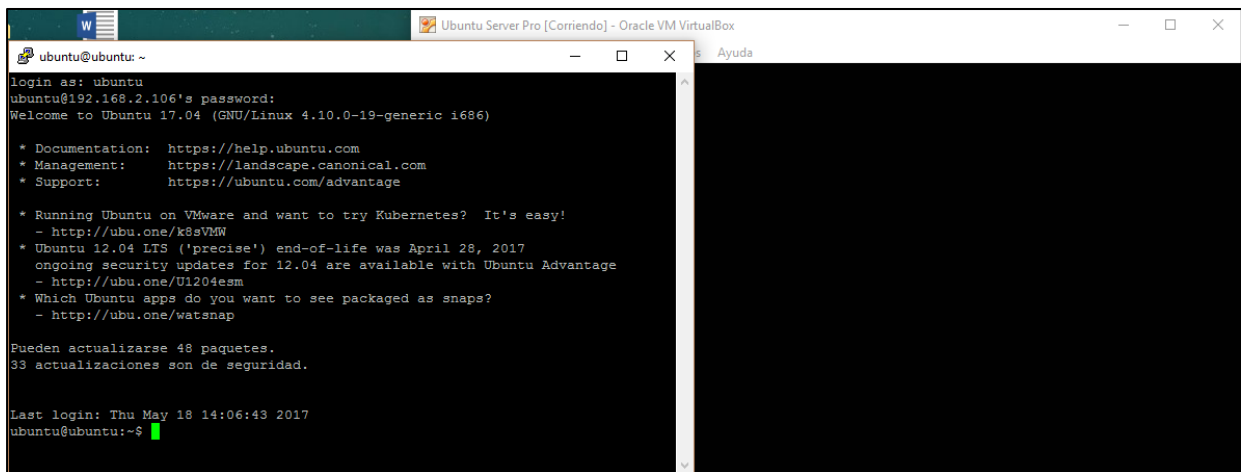
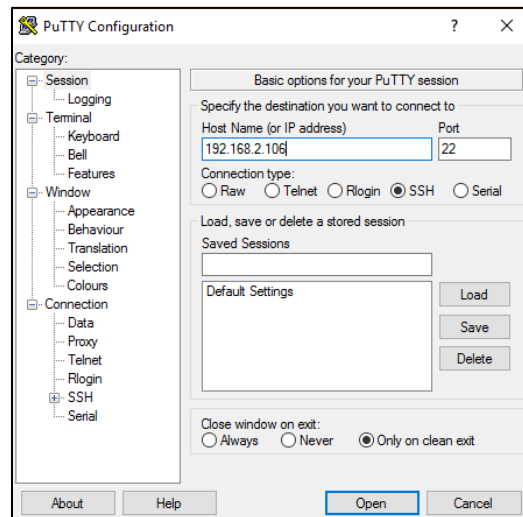
La interfaz gráfica Webmin también cuenta con este módulo de configuración de red, esta se explicará más adelante durante el desarrollo del anexo 1.



ACCESO SSH

Anteriormente se seleccionó el servidor SSH durante la instalación del Ubuntu Server 17.04, ahora describiremos el modo de acceso por SSH al servidor, cabe aclarar que el cliente debe tener conectividad bidireccional con el servidor, es decir, probar desde el cliente un “ping” exitoso con la IP del servidor.

Para este servicio utilizaremos un cliente con Windows 10; dado que es S.O con arquitectura distinta necesitaremos un programa interpretador de dicha arquitectura en LINUX como lo es PuTTY.



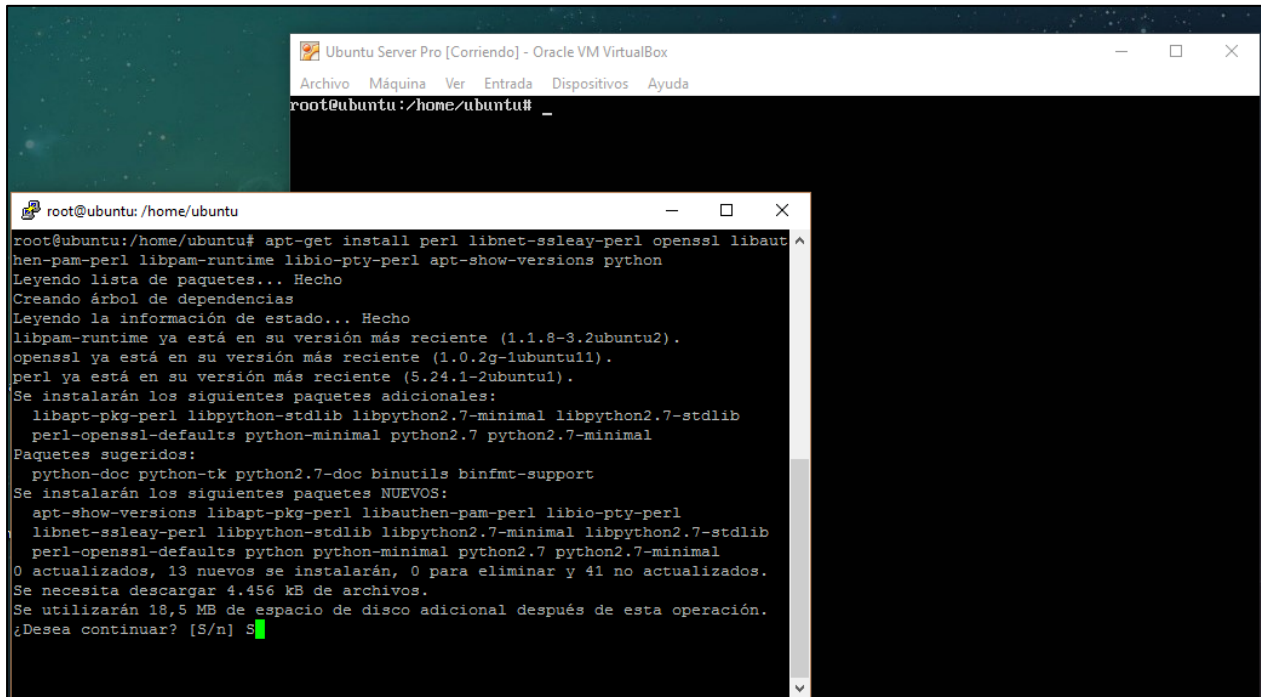
Este programa establecerá la conexión e iniciaremos sesión tal y como en el servidor. Una de las ventajas que tiene PuTTY respecto de la estructura del S.O anfitrión es que puedo copiar y pegar líneas de comandos o configuración desde el cliente, esto nos ahorra mucho tiempo a la hora de efectuar configuraciones ágiles.

Si nuestro cliente de acceso por SSH está basado en LINUX para obtener dicho acceso abrimos el terminal, y en modo súper usuario ejecutaremos el comando `ssh usuario@192.168.0.10` y enter, e iniciamos sesión como el caso anterior (la IP descrita corresponde a una IP ejemplo, esta IP debe ser la del servidor).

[Anexo 1]

SERVICIO DE ENTORNO GRÁFICO WEBMIN

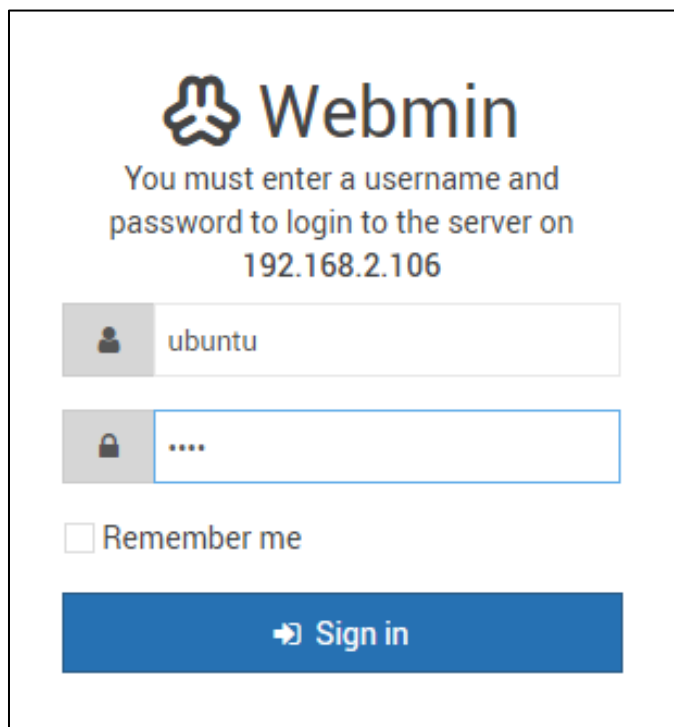
Webmin nos facilita la configuración y ejecución de algunos servicios de Ubuntu o sistemas basados en Debian, tiene un resultado equivalente al modo texto directamente en el servidor pero es más interactivo y fácil de interpretar. Para la instalación de este servicio descargaremos e instalaremos algunos paquetes desde su página oficial <http://www.webmin.com/deb.html> o bien ejecutaremos las siguientes líneas de comandos.



```
root@ubuntu:/home/ubuntu# apt-get install perl libnet-ssleay-perl openssl libauthen-pam-perl libpam-runtime libio-pty-perl apt-show-versions python
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
libpam-runtime ya está en su versión más reciente (1.1.8-3.2ubuntu2).
openssl ya está en su versión más reciente (1.0.2g-1ubuntu1).
perl ya está en su versión más reciente (5.24.1-2ubuntu1).
Se instalarán los siguientes paquetes adicionales:
  libapt-pkg-perl libpython2.7-minimal libpython2.7-stdlib
  perl-openssl-defaults python-minimal python2.7 python2.7-minimal
Paquetes sugeridos:
  python-doc python-tk python2.7-doc binutils binfmt-support
Se instalarán los siguientes paquetes NUEVOS:
  apt-show-versions libapt-pkg-perl libauthen-pam-perl libio-pty-perl
  libnet-ssleay-perl libpython-stdlib libpython2.7-minimal libpython2.7-stdlib
  perl-openssl-defaults python python-minimal python2.7 python2.7-minimal
0 actualizados, 13 nuevos se instalarán, 0 para eliminar y 41 no actualizados.
Se necesita descargar 4.456 kB de archivos.
Se utilizarán 18,5 MB de espacio de disco adicional después de esta operación.
¿Desea continuar? [S/n] S
```

```
#apt-get install perl libnet-ssleay-perl openssl libauthen-pam-perl libpam-runtime libio-pty-perl apt-show-versions Python
#wget http://prdownloads.sourceforge.net/webadmin/webmin_1.840_all.deb
# dpkg --install webmin_1.840_all.deb
```

En ese orden, es indispensable que el servidor tenga acceso a internet, una vez descargados e instalados podremos acceder a la interfaz de Webmin mediante un explorador web en el cliente dirigiéndonos a <https://192.168.0.14:10000/> donde la IP corresponde a la IP del servidor que se conecta por el puerto 10000. Si es necesario añadimos la excepción de seguridad en opciones avanzadas y continuamos.



Iniciaremos sesión tal y como lo haríamos en el S.O en modo texto y listo, la siguiente imagen es la primera impresión del modo gráfico Webmin con un Dashboard con varios parámetros de rendimiento del servidor, en la parte lateral izquierda tendremos los módulos y configuraciones. Webmin nos brinda la facilidad de hacer más dinámico y sencillo el proceso de configuración de servicios, es por esto que se empleó para la creación del servidor DNS el cual se describe en el anexo 2.

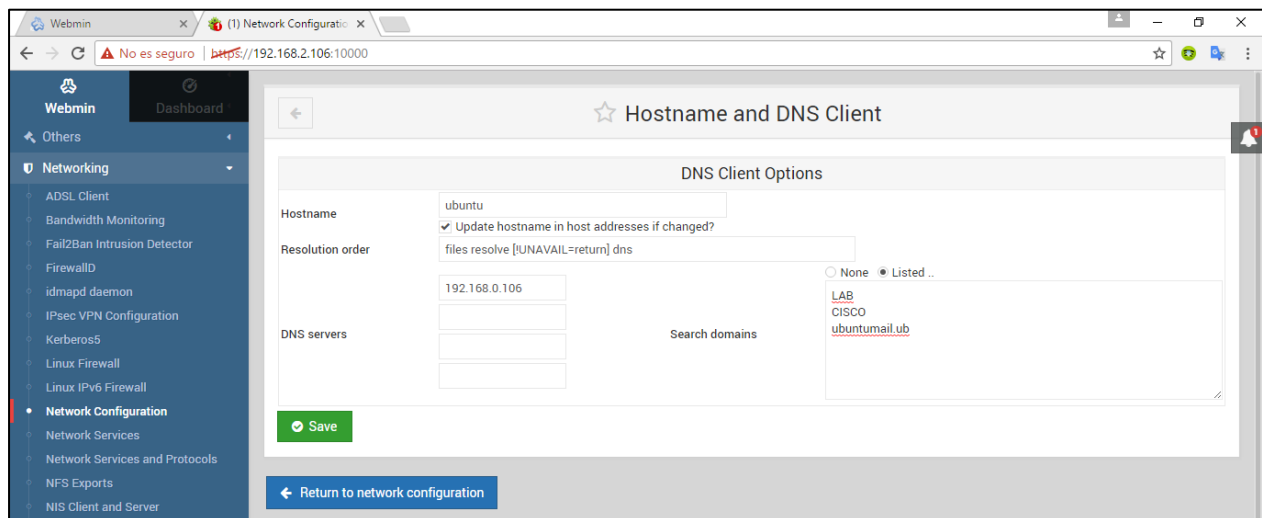
[Anexo 2]

SERVIDOR DNS

Considerándose el requisito planteado anteriormente en la creación y configuración de un servidor mail, tendremos que montar un servidor DNS con la dirección de nuestro servidor que interprete los protocolos www, mail, imap, pop3 y smtp, para el correcto funcionamiento de dicho servidor mail. Este servidor se configurará mediante la interfaz Webmin, recordemos que este servicio de DNS ya está instalado cuando instalamos el S.O pero aun no configurado con nuestro dominio para correo electrónico.

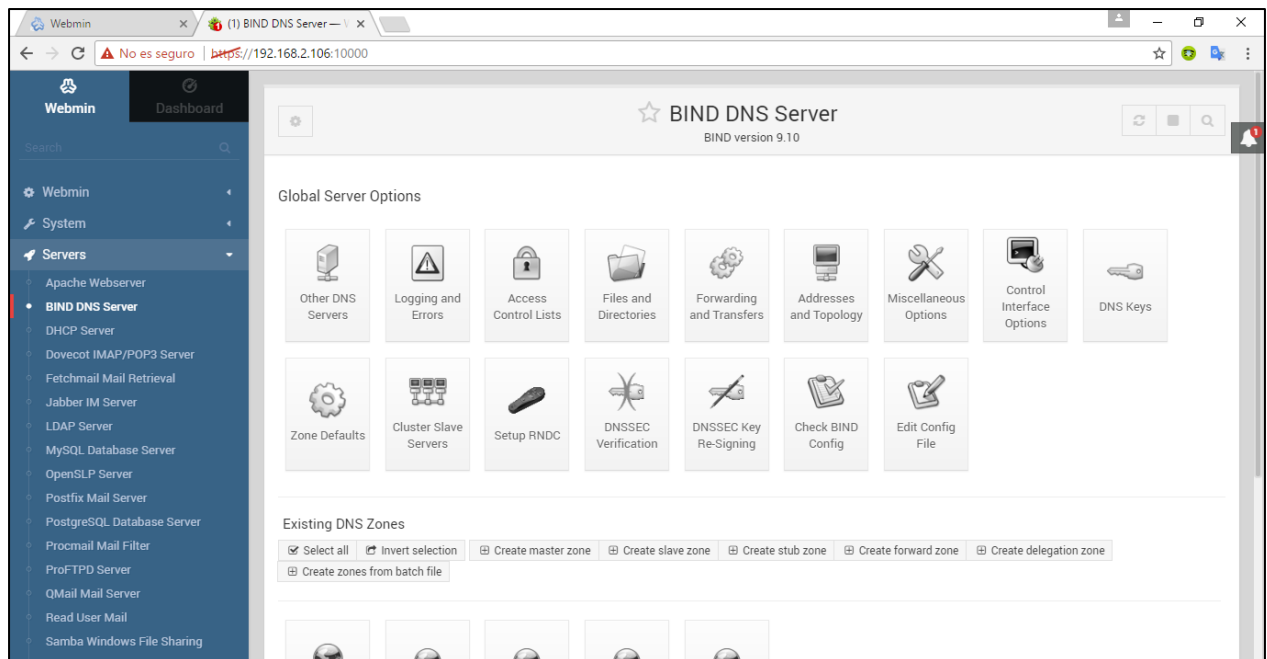
En primera instancia tendremos que tener dicho dominio asociado a la IP del servidor para elaborar la estructura de una dirección mail como por ejemplo **usuario1@midominio.com** de esta forma será fácil la interpretación de las direcciones mail para cualquier usuario que envíe o reciba email.

Se realizará el servidor DNS bajo las condiciones descritas anteriormente con el dominio **ubuntumail.ub** y la IP **192.168.2.106**.

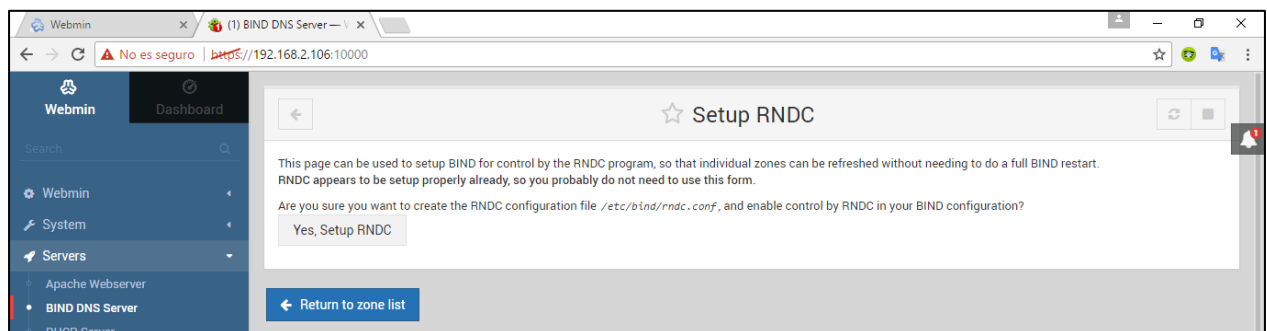


Una vez en Webmin tendremos que configurar nuestra red asignando la IP de nuestro servidor para la resolución de DNS de esa interfaz y listaremos el dominio **ubuntumail.ub**, aplicamos y salvamos los cambios.

Ya configurado este módulo, pasaremos a crear el dominio y configurarlo en el módulos DNS server, representado por el servicio **bind9** en el modo texto del nastro S.O anfitrión.

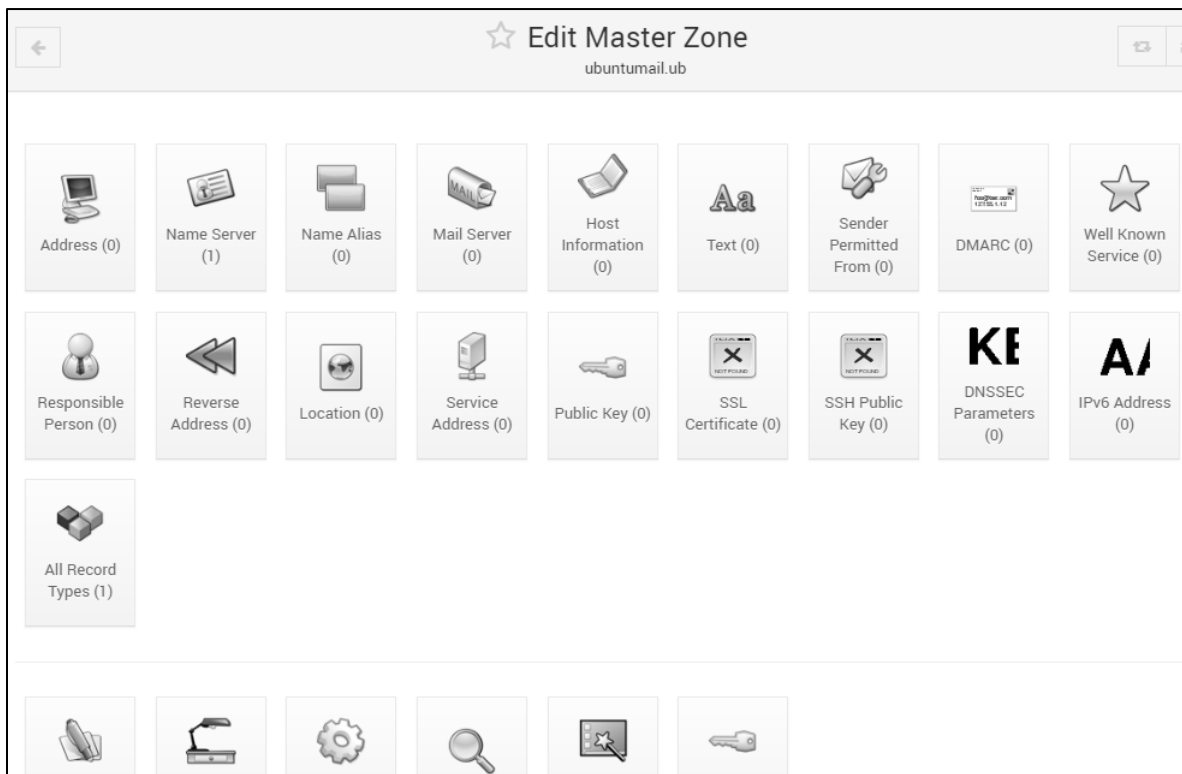


Aquí, antes que nada instalaremos el **RNDG** para poder crear y configurar nuestros dominios.



Instalado el RNDG procederemos al apartado anterior y crearemos una nueva zona maestra, dentro de esta zona pondremos en el modo normal y digitaremos nuestro dominio que queramos, en este ejemplo se creará el dominio `ubuntumail.ub`, podremos un email para la notificación de errores y ya está.

Una vez creada la zona de dominio nos enviará al apartado para editarla, aquí primero entraremos a address y lo configuraremos de la siguiente manera.



Address Records

In ubuntumail.ub

Add Address Record

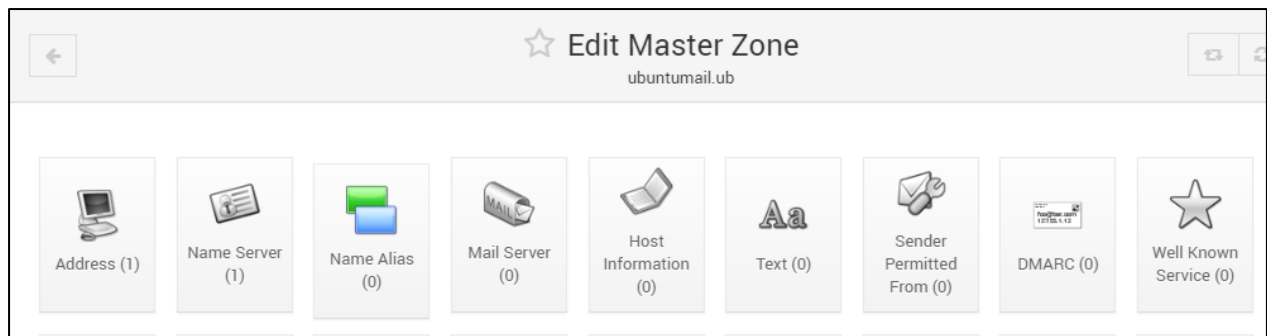
Name: Time-To-Live: ☒ Default ☐ seconds

Address:

Update reverse?: ☒ Yes ☐ Yes (and replace existing) ☐ No

Show records matching:

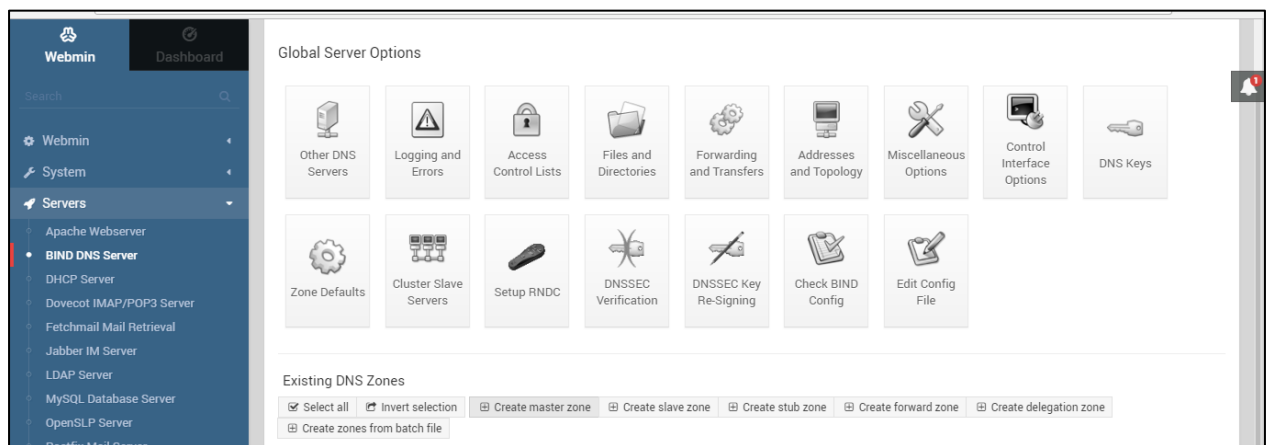
Aquí dentro pondremos en nombre el de nuestra máquina (el de la máquina no el del usuario de instalación, pero en nuestro caso resulta ser el mismo) y en la dirección la IP del servidor, creamos, aplicamos la zona en la parte superior derecha y volvemos al apartado anterior.



Ahora editaremos los alias, que serán los protocolos correspondido al servidor.

Name	TTL	Real Name
www.ubuntumail.ub	Default	ubuntu.ubuntumail.ub.
mail.ubuntumail.ub	Default	ubuntu.ubuntumail.ub.
smtp.ubuntumail.ub	Default	ubuntu.ubuntumail.ub.
imap.ubuntumail.ub	Default	ubuntu.ubuntumail.ub.

En nombre pondremos el nombre del protocolo y en nombre real pondremos el nombre de la máquina seguido de punto nombre de dominio y terminar con un punto (**ubuntu.ubuntumail.ub.**). Creamos y aplicamos, ahora volvemos al apartado de zonas.



Una vez más crearemos un zona maestra pero en configuración de zona inversa.

☆

Create Master Zone

New master zone options

Zone type

☐ Forward (Names to Addresses)
☒ Reverse (Addresses to Names)

Domain name / Network

192.168.0

Records file

☒ Automatic
☐

Master server

ubuntu

☒ Add NS record for master server?

Email address

root@ubuntumail.ub

Use zone template?

☐ Yes
☒ No

IP address for template records

Add reverses for template addresses?

☒ Yes
☐ No

Refresh time

10800

seconds

Transfer retry time

3600

seconds

Expiry time

604800

seconds

Negative cache time

38400

seconds

Create

Return to zone list

En la figura se evidencia que pondremos en lugar del nombre de dominio, la dirección de red de la LAN, una vez creada pasaremos a poner la dirección inversa como lo muestra la siguiente figura.

☆

Edit Master Zone

192.168.0

Reverse Address (1)

Name Server (1)

Name Alias (0)

All Record Types (2)

☆

Reverse Address Records

In 192.168.0

Add Reverse Address Record

Address

192.168.0.10

Time-To-Live

☒ Default
☐

seconds

Hostname

ubuntu.ubuntumail.ub.

Update forward?

☒ Yes
☐ No

Create

Show records matching:

Search

☒ Select all
☐ Invert selection

Address

TTL

Hostname

Ahora agregaremos la dirección que queremos que resuelva al hostmane como en el caso de zonas no inversas. Creamos, guardamos, aplicamos, cargamos los módulos y listo.

[24]

Pasaremos al servidor en su entorno de texto y reiniciaremos el servicio de DNS bajo el comando **service bind9 restart**, una vez ejecutado esto es aconsejable que se reinicie el servidor por completo usando **init 6**.

Reiniciado el servidor probaremos el nombre de dominio desde el mismo servidor y un cliente en Windows.

En el servidor ejecutaremos el comando **nslookup 192.168.0.10** (por ejemplo la IP del servidor) y la resolución para todos los demás protocolos, deberá mostrar esta forma.

```
root@ubuntu:/home/ubuntu# nslookup
> 192.168.0.10
Server:      192.168.0.10
Address:     192.168.0.10#53

10.0.168.192.in-addr.arpa      name = ubuntu.ubuntumail.ub.
> smtp
Server:      192.168.0.10
Address:     192.168.0.10#53

smtp.ubuntumail.ub           canonical name = ubuntu.ubuntumail.ub.
Name:   ubuntu.ubuntumail.ub
Address: 192.168.0.10
> imap
Server:      192.168.0.10
Address:     192.168.0.10#53

imap.ubuntumail.ub          canonical name = ubuntu.ubuntumail.ub.
Name:   ubuntu.ubuntumail.ub
Address: 192.168.0.10
> pop3
Server:      192.168.0.10
Address:     192.168.0.10#53

pop3.ubuntumail.ub          canonical name = ubuntu.ubuntumail.ub.
Name:   ubuntu.ubuntumail.ub
Address: 192.168.0.10
```

Probamos la IP del servidor resulta con el dominio ubuntu.ubuntumail.ub y los otros protocolos, la resolución de los otros protocolo permite que el servidor redireccione, por ejemplo www.ubuntumail.ub a ubuntu.ubuntumail.ub y así para el smtp, pop3 etc.

Ahora probaremos la resolución directa es decir no entrando por la IP si no por el dominio, este a su vez debe responder desde la IP del servidor bajo el protocolo www.

```
root@ubuntu:/home/ubuntu# nslookup
> www.ubuntumail.ub
Server:      192.168.0.10
Address:     192.168.0.10#53

www.ubuntumail.ub           canonical name = ubuntu.ubuntumail.ub.
Name:   ubuntu.ubuntumail.ub
Address: 192.168.0.10
>
```

Para hacer las prueba desde un cliente es claro resaltar que la resolución de dominios de la interfaz de red que use el cliente para conectarse a la LAN esté definida con la IP del servidor, una vez cambiado esto pasaremos a ejecutar la prueba simplemente haciendo ping a `www.ubuntumail.ub` y que el servidor nos dé respuesta desde su IP, es decir ahora al hacer ping el dominio será equivalente a hacerle ping directamente a la IP.

```

C:\> Símbolo del sistema

Microsoft Windows [Versión 10.0.14393]
(c) 2016 Microsoft Corporation. Todos los derechos reservados.

C:\Users\Jhossier>ping www.ubuntumail.ub

Haciendo ping a ubuntu.ubuntumail.ub [192.168.0.10] con 32 bytes de datos:
Respuesta desde 192.168.0.10: bytes=32 tiempo<1m TTL=64
Respuesta desde 192.168.0.10: bytes=32 tiempo=2ms TTL=64
Respuesta desde 192.168.0.10: bytes=32 tiempo=1ms TTL=64
Respuesta desde 192.168.0.10: bytes=32 tiempo=1ms TTL=64

Estadísticas de ping para 192.168.0.10:
    Paquetes: enviados = 4, recibidos = 4, perdidos = 0
    (0% perdidos),
    Tiempos aproximados de ida y vuelta en milisegundos:
    Mínimo = 0ms, Máximo = 2ms, Media = 1ms

C:\Users\Jhossier>
```

Evidenciamos que dicho dominio lo resuelve la IP del servidor. Ya con esto tendremos completado el servidor DNS para utilizarlo en la resolución de direcciones mail para los protocolos usados por el MUA desde el escritorio.

[Anexo 3]

SERVIDOR WEB APACHE 2

Este apartado contiene la configuración de un servidor web con soporte de PHP para el caso de que nuestro servicio de MUA sea vía web, para el desarrollo del ejemplo no fue nuestro caso pero es conveniente realizarlo ya que desconocemos las condiciones iniciales del lector.

Consideremos que ya tenemos creado y configurado nuestro dominio con soporte de protocolo www, el dominio ejemplo manejado hasta ahora es ubuntuemail.ub, por tanto el acople para soporte de protocolo se representaría de la forma www.ubuntuemail.ub.

Para empezar entraremos al Webmin e instalaremos el servidor web Apache.

Webmin Dashboard

Apache Webserver

The Apache server executable `/usr/sbin/apache2` does not exist. If you have Apache installed, adjust the [module configuration](#) to use the correct path. The Apache Webserver package can be automatically installed by Webmin. [Click here](#) to have it downloaded and installed using APT.

2.168.2.106:10000

Package	libaprutil1-dbd-sqlite3:i386	Class	K-O
Version	1.5.4-3	Vendor	Ubuntu Developers <ubuntu-devel-discuss@lists.ubuntu.com>
Architecture	i386	Installed	Unknown

Package apache2-utils was installed successfully

Package Details			
Description	Unknown		
Package	apache2-utils	Class	A-E
Version	2.4.25-3ubuntu2	Vendor	Ubuntu Developers <ubuntu-devel-discuss@lists.ubuntu.com>
Architecture	i386	Installed	Unknown

Package apache2-bin was installed successfully

Package Details			
Description	Unknown		
Package	apache2-bin	Class	A-E
Version	2.4.25-3ubuntu2	Vendor	Ubuntu Developers <ubuntu-devel-discuss@lists.ubuntu.com>
Architecture	i386	Installed	Unknown

Package apache2 was installed successfully

Package Details			
Description	Unknown		
Package	apache2	Class	A-E
Version	2.4.25-3ubuntu2	Vendor	Ubuntu Developers <ubuntu-devel-discuss@lists.ubuntu.com>
Architecture	i386	Installed	Unknown

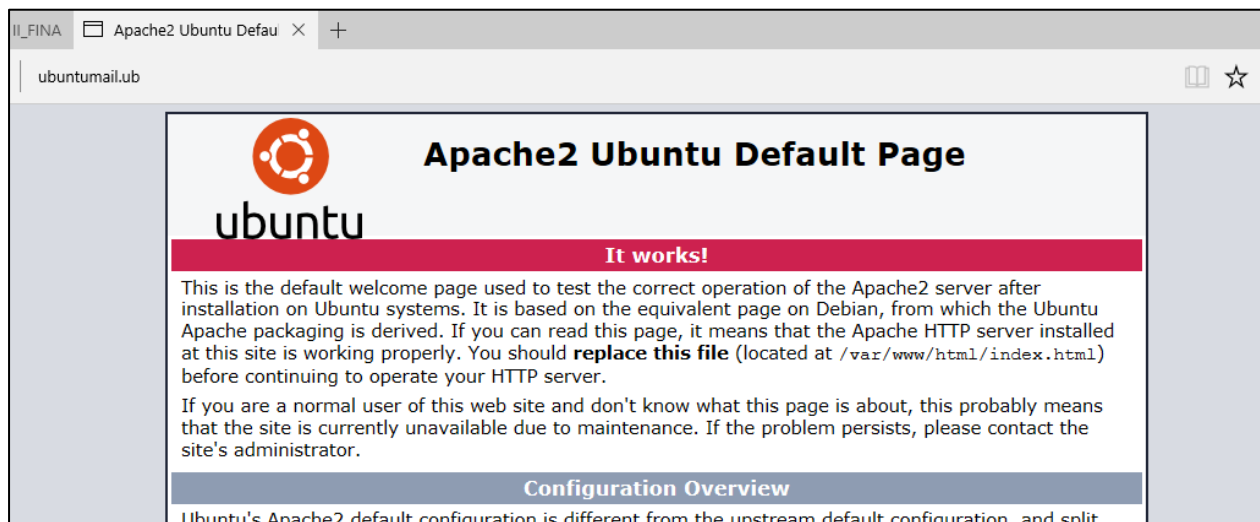
[Return to Apache Webserver](#)

Una vez realizada la instalación volveremos al módulo de configuración del servidor web Apache. Aquí en la pestaña de crear host virtual, crearemos el nuevo servidor web.

The screenshot shows the Apache Webserver configuration interface. At the top, it says 'Apache Webserver' and 'Apache version 2.4.25'. Below this, there are three tabs: 'Global configuration', 'Existing virtual hosts', and 'Create virtual host'. The 'Create virtual host' tab is selected. The main form is titled 'Create a New Virtual Server'. It contains several fields and options:

- Handle connections to address:** Radio buttons for 'Those not handled by another server', 'Any address', and 'Specific address ..'. The 'Specific address ..' option is selected, and the value '192.168.2.106' is entered in the adjacent text box.
- Listen on address (if needed):** A checked checkbox.
- Port:** Radio buttons for 'Default', 'Any', and '80'. The '80' option is selected.
- Document Root:** A text box with a file icon button next to it.
- Allow access to this directory:** A checked checkbox.
- Server Name:** Radio buttons for 'Automatic' and 'Standard httpd.conf file'. The 'Automatic' option is selected.
- Add virtual server to file:** Radio buttons for 'Standard httpd.conf file', 'New file under virtual servers directory /etc/apache2/sites-available', and 'Selected file..'. The 'New file under virtual servers directory /etc/apache2/sites-available' option is selected.
- Copy directives from:** A dropdown menu with 'Nowhere' selected.
- Create Now:** A green button with a plus icon.

Pondremos que nos dirija a una dirección IP específica y pondremos la IP del dominio, es decir, la IP del servidor, especificaremos el uso del puerto 80, que la página que cargue sea la de defecto ya que no necesitaremos configurar este apartado para nuestro ejemplo, pero, en caso de que el lector tenga una ruta de un archivo tipo index en el servidor para cargar su código HTML, puede buscarla aquí. Con esto se terminaría la configuración del servidor web, ahora procederemos a las pruebas.



Desde un cliente conectado a la LAN del servidor escribiremos el dominio en el motor de direcciones de su navegador de preferencia, se debe tener respuesta del servidor con la página de defecto de Apache 2 o de su página en caso de que la haya cargado, de esta forma tenemos el servidor web ejecutándose.