



UNIVERSIDAD LUTERANA SALVADOREÑA

FACULTAD DE CIENCIAS DEL HOMBRE Y LA NATURALEZA

LICENCIATURA DE CIENCIAS DE LA COMPUTACIÓN

ING. MANUEL FLORES VILLATORO

REDES I

SERVIDOR DE CORREOS

INTEGRANTES				
N°	CARNET	APELLIDOS	NOMBRES	PARTICIPACIÓN
1	AA01132424	ACOSTA ARIAS	JOSUÉ ISAAC	80%
2	DP02110390	DIAZ PALACIOS	RAFAEL ANTONIO	90%
3	VY01132301	VEGA YANES	MIGUEL ALEJANDRO	80%

INDICE	
INTRODUCCIÓN	3
OBJETIVOS	4
GENERAL:	4
ESPECÍFICOS:	4
MARCO TEÓRICO.....	5
DESCRIPCIÓN DEL PROYECTO	11
DIAGRAMA DE GANTT	12
DIAGRAMA DE RED.....	13
FACTIBILIDAD DEL PROYECTO	14
CONCLUSIÓN	27
BIBLIOGRAFÍA	28

Índice de Tablas

FACTIBILIDAD DEL PROYECTO	14
---------------------------------	----

Índice de ilustraciones

DIAGRAMA DE GANTT	12
DIAGRAMA DE RED.....	13

INTRODUCCIÓN

El documento tiene como propósito describir el proyecto que como grupo elegimos

El proyecto se trata de implementar un Servidor de Correo.

Para iniciar con el proyecto hemos investigado un poco acerca de servidor de correos y el diferente software para GNU/LINUX que podemos implementar.

El correo electrónico es considerado el servicio más utilizado de Internet. Por lo tanto, la serie de protocolos TCP/IP ofrece una gama de protocolos que permiten una fácil administración del enrollamiento del correo electrónico a través de la red.

Toda comunicación entre ordenadores debe regirse por un protocolo, y existen dos

Protocolos para el acceso desde PCs al correo:

- POP (Post Office Protocol)
- IMAP (Internet Messsage Access Protocol)

Estos protocolos nos permiten acceder a nuestros mensajes almacenados en un servidor. Es

OBJETIVOS.

GENERAL:

✓ Mostrar el funcionamiento de un Servidor de Correos , describir los diferentes paquetes y protocolos utilizados para su implementación.

ESPECÍFICOS:

✓ Realizar una demostración del funcionamiento del Servidor de Correos.

✓ Conocer sobre protocolos de correos.

✓ Verificar los requerimientos mínimos de hardware y software para funcionamiento de un servidor de correos.

MARCO TEÓRICO

1. Servidor de correos.

Un servidor de correo es una aplicación de red ubicada en un servidor en internet.

El MTA (Mail Transfer Agent) tiene varias formas de comunicarse con otros servidores de correo:

1. Recibe los mensajes desde otro MTA. Actúa como "servidor" de otros servidores.
2. Envía los mensajes hacia otro MTA. Actúa como un "cliente" de otros servidores.
3. Actúa como intermediario entre un "Mail Submission Agent" y otro MTA.

1.2 Intercambio de correo electrónico.

Un servidor de correo realiza una serie de procesos que tienen la finalidad de transportar información entre los distintos usuarios. Usualmente el envío de un correo electrónico tiene como fin que un usuario (remitente) cree un correo electrónico y lo envíe a otro (destinatario). Esta acción toma típicamente 5 pasos:

1. El usuario inicial crea un "correo electrónico"; un archivo que cumple los estándares de un correo electrónico. Usará para ello una aplicación ad-hoc. Las aplicaciones más usadas, en indistinto orden son: Outlook Express Microsoft), Microsoft Outlook, Mozilla Thunderbird (Mozilla), Pegasus Mail (David Harris), Lotus Notes IBM), etc.
2. El archivo creado es enviado a un almacén; administrado por el servidor de correo local al usuario remitente del correo; donde se genera una solicitud de envío.
3. El servicio MTA local al usuario inicial recupera este archivo e inicia la negociación con el servidor del destinatario para el envío del mismo.
4. El servidor del destinatario valida la operación y recibe el correo, depositándolo en el "buzón" correspondiente al usuario receptor del correo. El "buzón" no es otra cosa que un registro en una base de datos.
5. Finalmente el software del cliente receptor del correo recupera este archivo o "correo" desde el servidor almacenando una copia en la base de datos del

programa cliente de correo, ubicada en la computadora del cliente que recibe el correo.

2. Cliente de correo electrónico.

Un cliente de correo electrónico es un programa de ordenador usado para leer y enviar mensajes de correo electrónico.

Originalmente, los clientes de correo electrónico fueron pensados para ser programas simples para leer los mensajes del correo de usuario, enviados por el agente de reparto de correo (MDA) conjuntamente con el agente de transferencia de correo (MTA) a un buzón local.

Los formatos de buzón de correo más importantes son Imbox y Maildir. Estos simplísimos protocolos para el almacenamiento local de los mensajes de correo electrónico realizan de una forma muy sencilla la importación, exportación y copia de seguridad de las carpetas de correo.

Los mensajes de correo electrónico pendientes de envío serán entregados al MTA, tal vez a través de un agente de correo saliente, de forma que el cliente de correo electrónico no necesita proporcionar ninguna clase de función de transporte.

3. Protocolos para el intercambio de correos

Para el intercambio de mensajes entre personas (y archivos adjuntos como imágenes, documentos, de texto, etc.), el servicio de correo electrónico se sirve de diversos protocolos. Estos protocolos permiten que máquinas distintas, que se ejecutan con frecuencia en sistemas operativos y con programas de correo electrónico diferentes, se comuniquen entre sí e intercambien mensajes para que lleguen a los destinatarios adecuados (Red Hat Linux 2002).

Podemos hablar de dos tipos de protocolos: los que le van a permitir a un usuario acceder a su buzón de mensajes en un servidor, y los que le van a permitir enviar mensajes a otros usuarios. En el primer grupo, los dos protocolos más populares son IMAP (Internet Message Access Protocol, Protocolo de Acceso a Mensajes de Internet) y POP (Post Office Protocol, Protocolo de Oficina de Correo). La principal diferencia

reside en que el protocolo IMAP permite el acceso a los mensajes alojados en el servidor y POP la descarga en la máquina local, borrándolos o dejando una copia en el servidor, según se indique.

POP fue diseñado inicialmente para leer correos sin conexión. El usuario se conectaba y descargaba los correos a su máquina local después de lo cual éstos eran borrados del servidor. La principal desventaja de esta forma de operación era que no era compatible con el acceso desde múltiples servidores, porque tendía a dispersar el correo por todas las máquinas desde las cuales se revisara. Así, el modo de acceso “sin conexión” ataba a los usuarios a usar un equipo para el almacenamiento y manipulación de mensajes.

IMAP en cambio, fue pensado para permitir el acceso y la gestión de los mensajes desde más de un computador. Además soportaba modos de acceso “en línea”, “sin conexión” y “desconectado”; accesos concurrentes a buzones de correo compartidos; y fue pensado para ser completamente compatible con estándares de mensajería en Internet como MIME (Multipurpose Internet Mail Extensions, Extensiones Multipropósito de Correo en Internet). En cuanto al segundo grupo, tenemos en él al protocolo SMTP (Simple Mail Transfer Protocol), Protocolo simple de transferencia de correo), descrito en el RFC 821.

IRedMail

Para la llevar a cabo el proyecto de Servidor de Correo utilizamos el script instalador IRedMail.

Definición de IRedMail:

Es una plataforma robusta y completa para administrar sistemas de correo (mail), y que permite instalar y configurar todos los paquetes que se necesitan para hacer funcionar un Servidor de Correo, es decir, que soporte todo en uno:

- Servidor SMTP
- Autenticación
- Servidor POP3/IMAP

- Anti-Spam (RBL)
- Anti-virus (Clamav)
- Soporte MySQL, PostgreSQL, openLDAP
- Dominios Virtuales (virtual domains)
- Webmail
- Full administración vía Web.

POSTFIX

Es un Agente de Transporte de Correo (MTA) de código abierto, o lo que es lo mismo, un programa informático para el enrutamiento y envío de correo electrónico.

Postfix fue creado como alternativa a Sendmail, buscando un servidor que fuera más rápido, fácil de administrar y seguro. Postfix es de hecho, el MTA que se usa por defecto en muchos sistemas operativos derivados de UNIX, entre ellos, GNU/Linux.

Fue desarrollado por Wietse Venema durante una estancia en el Centro de Investigación Thomas J. Watson de IBM y fue conocido con el nombre de Mailer e IBM Secure Mailer, siendo distribuido al gran público por primera vez a mediados de 1999. Desde entonces sigue en proceso de desarrollo y mejora de una forma activa.

Características

Las principales características de Postfix son:

- ✓ Soporte para TLS (Transport Layer Security).
- ✓ Soporte para distintas bases de datos LDAP, MySQL, PostgreSQL.
- ✓ Soporte para mbox, maildir y dominios virtuales.
- ✓ SMTP-AUTH, SASL y reescritura de dirección.
- ✓ Soporte para Militer.
- ✓ Capacidad de manejar altos volúmenes de correo.

Algunas de la Principales Virtudes de Postfix son:

- ✓ Gran Seguridad: Desde el comienzo de su diseño ésta ha sido una de sus premisas fundamentales.

Características:

1. Seguridad frente a ataques contra el servidor.

2. Seguridad contra el uso inadecuado (spam, relay, etc.), debido a que

Soporta directamente (sin modificaciones suplementarias) listas negras y que es complicado configurarlo como relay abierto. Además, se puede instalar Postfix de forma que corra en modo chroot, lo que le confiere a su operativa más seguridad.

✓ **Gran Rendimiento:** Postfix puede procesar cientos de miles o millones de mensajes al día sin problemas (algo que le desmarca de Sendmail). De hecho, según la documentación, un "PC normal" puede recibir y entregar hasta un millón de mensajes distintos al día con Postfix. Esta potencia se debe en parte a su modularidad, que además viene con el añadido de que se pueden definir ciertos parámetros para cada uno de los procesos, como el número máximo de procesos simultáneos de un tipo, activar o desactivar un proceso (funcionalidad) innecesario, etc., que permiten optimizar aún más su funcionamiento. Además, el sistema de gestión de colas de mensajes es también modular, consistiendo en 4 colas distintas que está procesadas muy eficientemente.

➤ **Soporte para las tecnologías más actuales:** Al estar muy activamente actualizado, emplea técnicas desarrolladas para aprovechar mejor y dar soporte a los servidores Web más modernos. Postfix soporta:

- LDAP

- Bases de datos (MySQL),.

- Autenticación mediante SASL, LMTP, etc.

✓ Muy buen soporte para administrar dominios virtuales.

✓ **Facilidad de configuración:**

- Con los dos ficheros de configuración: main.cf y master.cf se controla todo, y además están muy bien explicados.

- Uso sencillo de listas negras.

✓ Abundante documentación, y de calidad: Aunque en su mayoría, en inglés.

✓ **Fácil integración con programas antivirus:** Debido a que se pueden insertar procesos externos entre ciertas partes del sistema Postfix, lo cual es muy útil para integrar un

antivirus. Un ejemplo de ello es ClamAV (véase artículo configuración de un servidor incluyendo Postfixadmin, Mysql, Spamassassin y ClamAv).

✓ **Facilidad para detectar errores:** Postfix tiene múltiples formas de obtener información de los errores ocurridos y los logs generados son muy claros y explicativos.

Además, gracias a su modularidad es más sencillo saber qué proceso es el que falla, además de que se puede activar la emisión de más información de depuración de forma independiente para cada programa.

✓ **Posibilidad de lanzar varias instancias de Postfix en la misma máquina con distintas configuraciones:** Podemos usar cada una de estas instancias con distintas direcciones IP, distintos puertos, etc. De esta forma podemos tener más de un servidor para cada necesidad.

✓ **Utilidades:** Incluye utilidades para varias tareas, como por ejemplo, para gestionar las colas de mensajes. Otra ventaja conferida en parte por su modularidad.

✓ **Código Fuente abierto y bien estructurado:** Se considera el código fuente de Postfix como un ejemplo de diseño, claridad y documentación, por lo que facilita su mantenimiento por parte de desarrolladores así como la incorporación de nuevas capacidades, corrección de errores, adaptaciones, etc.

DESCRIPCIÓN DEL PROYECTO

El Proyecto a realizar consiste, en la implementación de un servidor de correos. Como primer instancia se llevara a cabo las investigaciones sobre la estructura lógica del desarrollo, implementación y configuraciones necesarias de un servidor de correo en el Sistema Operativo Debian. Para el desarrollo del proyecto, instalamos postfix el cual nos permitió realizar las configuraciones de forma simple y con menor tiempo.

También investigamos sobre algunos protocolos que son de suma importancia para el transporte del email, como lo son SMTP, IMAP y POP.

El protocolo SMTP, es el que nos permitirá la transferencia simple de correos electrónicos, es decir funcionará para realizar el intercambio del mensaje entre computadora cualquier otro dispositivo. IMAP, es el protocolo que nos permitirá obtener el mensaje electrónico a clientes locales. POP, este protocolo de red de acceso a mensajes electrónicos almacenados en un servidor.

DIAGRAMA DE GANTT

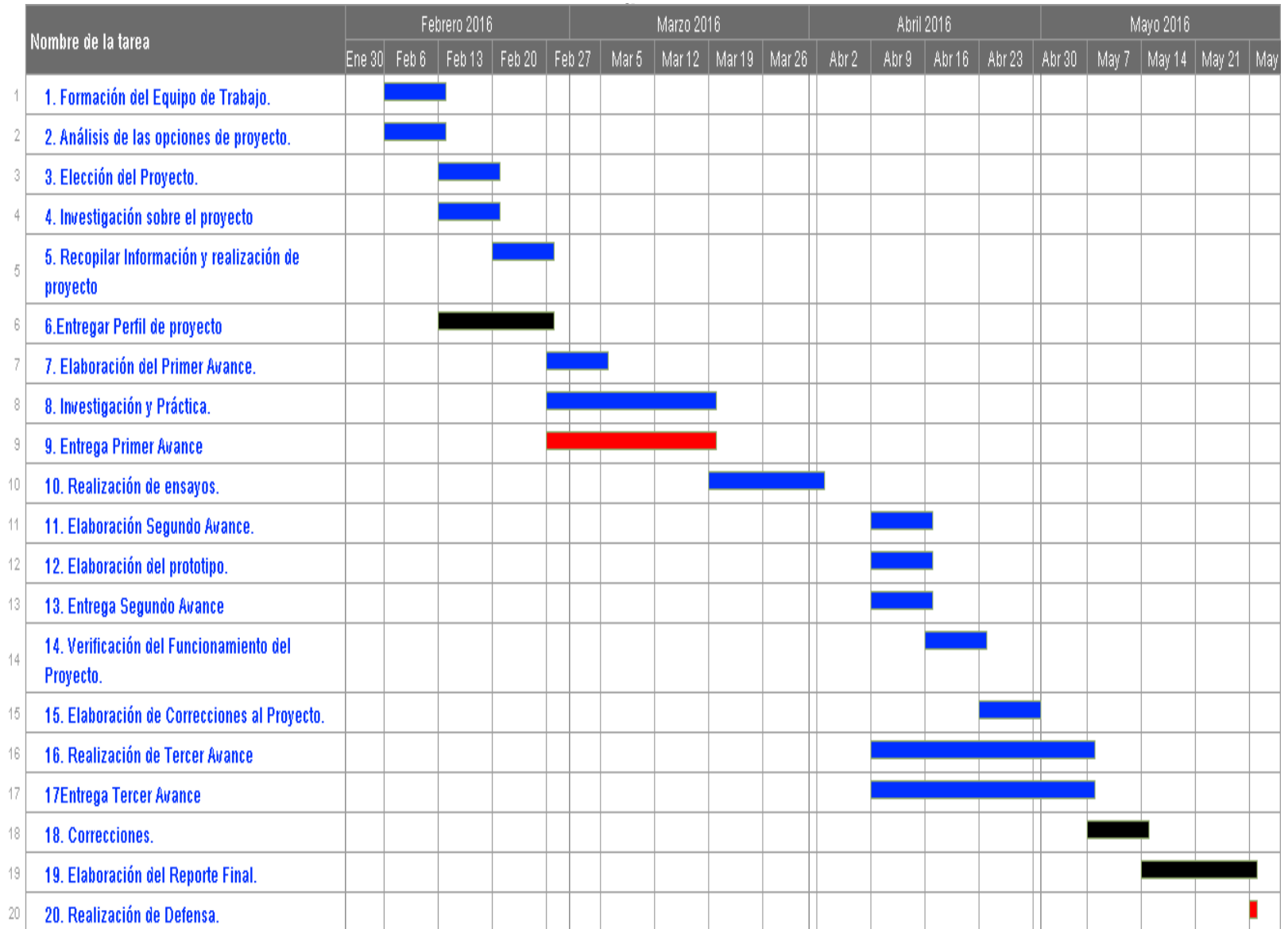
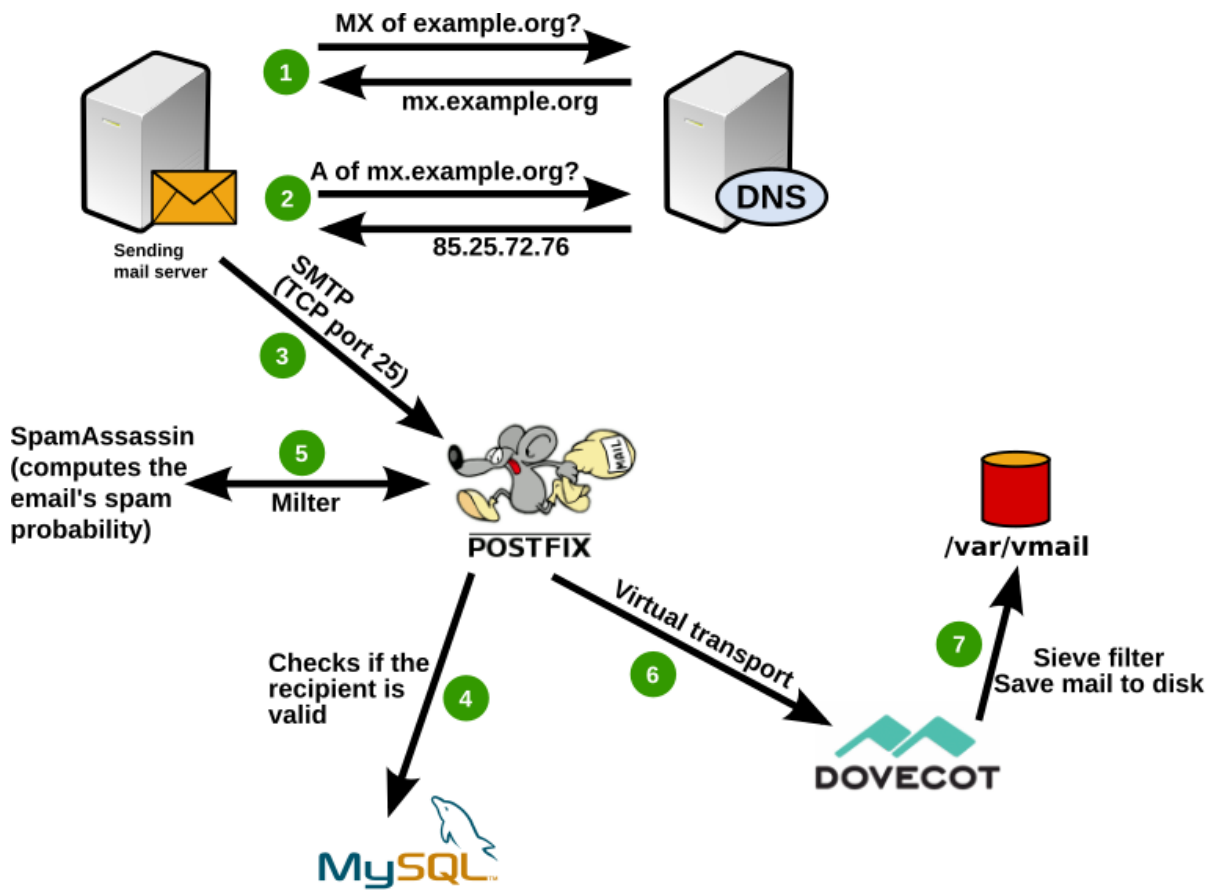


DIAGRAMA DE RED.



FACTIBILIDAD DEL PROYECTO

Al tener seleccionado el proyecto que desarrollaremos durante el ciclo, es pertinente realizar un estudio de factibilidad, los cuales permitirán determinar los aspectos importantes que se tomarán en cuenta para el desarrollo del proyecto, que es “Implementación de un Servidor de Correo”.

⇒ Factibilidad Técnica.

Para la Implementación de un Servidor de Correo, se requiere tanto de protocolos, como de diversos paquetes que tienen los requerimientos técnicos para el desarrollo y puesta en marcha del proyecto.

Se llevo a cabo la evaluación necesaria tanto de Hardware y Software que se utilizará para el desarrollo del proyecto.

En cuanto al Hardware:

- ⇒ Computadora servidor
- ⇒ Internet
- ⇒ Disco Duro

En cuanto al Software:

- ⇒ Sistema Operativo: Debian Jessie
- ⇒ Paquete Postfix.

De acuerdo al software y hardware, determinamos el proyecto es factible en la parte operativa del proyecto.

⇒ Factibilidad Económica.

A continuación se presenta los costos de los recursos, que se determinaron se utilizarán para el desarrollo del proyecto.

Detalle	Cantidad	Costo
PC con servidor	1	\$ 200.00
S.O GNU/Linux (Debian)	1	\$ 0.00
Software Postfix	1	\$ 0.00
Mano de Obra	3	\$ 50.00
Total		\$250.00

Dentro de la factibilidad económica se describió uno de los beneficios que permitirá en gran medida realizar el proyecto; también los paquetes de Postfix que al descargarlos no tendrán ningún costo el uso de su tecnología. Nuestra finalidad es buscar el funcionamiento del servidor de correo, ya que al funcionar tendremos la oportunidad de mostrar a los demás, la forma de enviar correos a través del servidor de correo.

⇒ Factibilidad Operativa.

El objetivo que como grupo tenemos, es lograr la implementación del servidor de correo.

DESARROLLO.

Configuración DNS

Es necesario configurar nuestro DNS para que resuelva nuestro correo llegue a nuestro equipo.

Para ello configuraremos nuestro servidor DNS, en esta fase del proyecto realizaremos las pruebas desde nuestro equipo local.

En el caso de no tener ningún servidor DNS configurado en nuestra red o de querer utilizar el servidor de correo solo localmente, podemos hacer las modificaciones del nombre de dominio en el archivo hosts situado en /etc/hosts.

```
root@debianpc:/home/rafaga8# ping casita.com
PING casita.com (192.168.1.12) 56(84) bytes of data.
64 bytes from 192.168.1.12: icmp_seq=1 ttl=64 time=0.012 ms
64 bytes from 192.168.1.12: icmp_seq=2 ttl=64 time=0.019 ms
64 bytes from 192.168.1.12: icmp_seq=3 ttl=64 time=0.026 ms
64 bytes from 192.168.1.12: icmp_seq=4 ttl=64 time=0.023 ms
64 bytes from 192.168.1.12: icmp_seq=5 ttl=64 time=0.016 ms
64 bytes from 192.168.1.12: icmp_seq=6 ttl=64 time=0.025 ms
64 bytes from 192.168.1.12: icmp_seq=7 ttl=64 time=0.024 ms
```

Para comprobar el funcionamiento podemos realizar un ping a nuestro nombre de dominio. Para el ejemplo usaremos el nombre de “casita.com”

Instalación de Postfix.

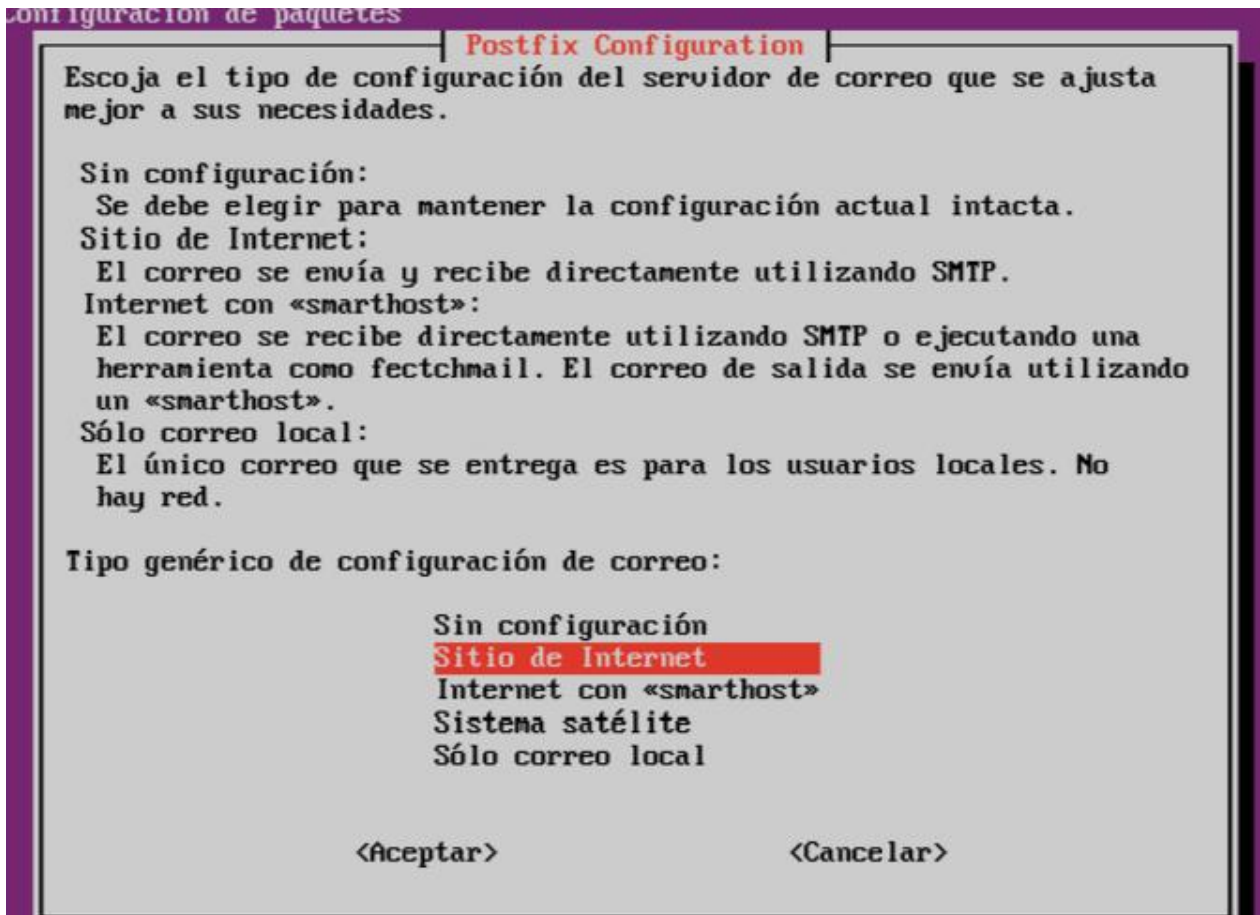
Para instalar Postfix, introduciremos el siguiente comando:

```
# aptitude install postfix
```

Configuración durante la instalación

Durante la instalación de Postfix, nos aparecen algunas pantallas de configuración.

La primer pantalla nos pide que seleccionemos un tipo de configuración para nuestro servidor de correos. Nosotros escogimos "sitio de internet".



En la siguiente pantalla, nos pide que escribamos nuestro FQDN (Fully Qualified Domain Name), , nuestro FQDN sera casita.com.

Una vez instalado Postfix, podemos acceder al fichero de configuración principal `/etc/postfix/main.cf` donde añadiremos al final las dos lineas que podemos ver a configuración:

```
inet protocols = ipv4
```

```
home mailbox = Maildir/
```

Especificando el protocolo de red utilizado (IPv4) y el directorio donde deben guardarse los correos electrónicos.

La parte final del fichero quedara pues de la siguiente manera:

```
rafaga8@debianpc: ~
Archivo  Editar  Ver  Buscar  Terminal  Pestañas  Ayuda
rafaga8@debianpc: ~ x rafaga8@debianpc: ~ x
GNU nano 2.2.6 Fichero: /etc/postfix/main.cf Modificado
myhostname = debianpc
alias_maps = hash:/etc/aliases
alias_database = hash:/etc/aliases
myorigin = /etc/mailname
mydestination = casita.com, debianpc, localhost.localdomain, localhost
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/
^G Ver ayuda ^O Guardar ^R Leer Fich ^Y Pág Ant ^K CortarTxt ^C Pos actual
^X Salir ^J Justificar ^W Buscar ^V Pág Sig ^U PegarTxt ^T Ortografía
```

Reiniciado de Postfix

A continuación reiniciaremos Postfix:

```
# /etc/init.d/postfix restart
```

Instalación de paquetes

```
apt-get install postfix postfix-mysql dovecot-core dovecot-imapd dovecot-lmtpd dovecot-mysql
```

Creación de la de bases de datos, para el servidor de correos.

```
mysqladmin -p create servermail
```

Creación de permisos y asignación de contraseña a la base de datos.

```
mysql > GRANT SELECT ON servermail.* TO 'usermail'@'127.0.0.1' IDENTIFIED BY 'mailpassword';
```

Aplicamos los cambios

```
mysql > FLUSH PRIVILEGES;
```

Usamos la base de datos servermail

```
mysql> USE servermail;
```

Crearemos 3 tablas.

La de dominios virtuales.

La tabla de alias

y la tabla de usuarios.

Script para la creación de dominios

```
CREATE TABLE `virtual_domains` (  
  `id` INT NOT NULL AUTO_INCREMENT,  
  `name` VARCHAR(50) NOT NULL,  
  PRIMARY KEY (`id`)  
  ) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

script para la creación de cuentas de usuarios.

```
CREATE TABLE `virtual_users` (  
  `id` INT NOT NULL AUTO_INCREMENT,  
  `domain_id` INT NOT NULL,  
  `password` VARCHAR(106) NOT NULL,  
  `email` VARCHAR(120) NOT NULL,  
  PRIMARY KEY (`id`),
```

```
UNIQUE KEY `email` (`email`),  
FOREIGN KEY (domain_id) REFERENCES virtual_domains(id) ON DELETE CASCADE  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

creación de alias virtuales, para las cuentas de correo.

```
CREATE TABLE `virtual_aliases` (  
  `id` INT NOT NULL AUTO_INCREMENT,  
  `domain_id` INT NOT NULL,  
  `source` varchar(100) NOT NULL,  
  `destination` varchar(100) NOT NULL,  
  PRIMARY KEY (`id`),  
  FOREIGN KEY (domain_id) REFERENCES virtual_domains(id) ON DELETE CASCADE  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

Ingreso de dominio a la base de datos

```
INSERT INTO `servermail`.`virtual_domains`  
(`id`, `name`)  
VALUES  
(1, 'manttoreales.com');
```

Ingreso de cuentas de correo.

```
INSERT INTO `servermail`.`virtual_users`  
(`id`, `domain_id`, `password`, `email`)  
VALUES  
(1, 1, ENCRYPT('manttoreales', CONCAT('$6$', SUBSTRING(SHA(RAND()), -16))),  
'manttoreales@manttoreales.com'),  
(2, 1, ENCRYPT('manttoreales', CONCAT('$6$', SUBSTRING(SHA(RAND()), -16))),  
'ventas@manttoreales.com');
```

salimos de mysql

```
mysql > exit
```

Editamos el archivo de postfix, main.cf.

```
nano /etc/postfix/main.cf
```

Editamos las siguientes líneas

```
virtual_transport = lmtp:unix:private/dovecot-lmtp
```

Acá indicamos donde se encuentran los archivos que hacen las consultas a la base de datos, para obtener la información.

```
virtual_mailbox_domains = mysql:/etc/postfix/mysql-virtual-mailbox-domains.cf
```

```
virtual_mailbox_maps = mysql:/etc/postfix/mysql-virtual-mailbox-maps.cf
```

```
virtual_alias_maps = mysql:/etc/postfix/mysql-virtual-alias-maps.cf
```

Creamos los 3 archivos agregados al main.cf de postfix, con el siguiente contenido

```
nano /etc/postfix/mysql-virtual-mailbox-domains.cf
```

```
este archivo consulta los dominios creados.
```

```
user = usermail
```

```
password = mailpassword
```

```
hosts = 127.0.0.1
```

```
dbname = servermail
```

```
query = SELECT 1 FROM virtual_domains WHERE name='%s'
```

podemos probar el fichero creado con el comando

```
postmap -q example.com mysql:/etc/postfix/mysql-virtual-mailbox-domains.cf
```

Ahora hacemos el archivo que obtendrá las cuentas de usuarios.

```
nano /etc/postfix/mysql-virtual-mailbox-maps.cf
```

```
user = usermail
password = mailpassword
hosts = 127.0.0.1
dbname = servermail
query = SELECT 1 FROM virtual_users WHERE email='%s'
```

igual lo probamos con el siguiente código.

```
postmap -q email1@example.com mysql:/etc/postfix/mysql-virtual-mailbox-maps.cf
```

y ahora creamos el archivo que nos dará los alias.

```
nano /etc/postfix/mysql-virtual-alias-maps.cf
```

```
user = usermail
password = mailpassword
hosts = 127.0.0.1
dbname = servermail
query = SELECT destination FROM virtual_aliases WHERE source='%s'
```

de igual manera lo probamos.

```
postmap -q alias@example.com mysql:/etc/postfix/mysql-virtual-alias-maps.cf
```

reiniciamos postfix.

```
service postfix restart
```

Ahora integramos dovecot con postfix.

En el dovecot.conf

```
nano /etc/dovecot/dovecot.conf
```

descomentamos esta linea

```
!include conf.d/*.conf  
!include_try /usr/share/dovecot/protocols.d/*.protocol
```

agregamos los protocolos que vamos a usar.

```
protocols = imap lmtp
```

definimos la localización donde se guardaran las bandejas de los correos.

```
nano /etc/dovecot/conf.d/10-mail.conf
```

```
mail_location = maildir:/var/mail/vhosts/%d/%n
```

```
mail_privileged_group = mail
```

creamos la carpeta de /var/mail y verificamos los permisos.

```
ls -ld /var/mail
```

```
drwxrwsr-x 3 root vmail 4096 Jan 24 21:23 /var/mail
```

creamos la carpeta de vhost y damos los permisos al usuario vmail.

```
mkdir -p /var/mail/vhosts/example.com
```

```
groupadd -g 5000 vmail
```

```
useradd -g vmail -u 5000 vmail -d /var/mail
```

```
chown -R vmail:vmail /var/mail
```

Configuracion el archivo de autenticacion de dovecot.

```
nano /etc/dovecot/conf.d/10-auth.conf
```

```
disable_plaintext_auth = yes
```

```
auth_mechanisms = plain login
```

```
#!include auth-system.conf.ext
```

```
!include auth-sql.conf.ext
```

configuracion el archivo de autenticacion a la base sql por dovecot.

```
nano /etc/dovecot/conf.d/auth-sql.conf.ext
```

```
passdb {  
    driver = sql  
    args = /etc/dovecot/dovecot-sql.conf.ext  
}  
userdb {  
    driver = static  
    args = uid=vmail gid=vmail home=/var/mail/vhosts/%d/%n  
}
```



```
nano /etc/dovecot/dovecot-sql.conf.ext
```

```
driver = mysql
```

```
connect = host=127.0.0.1 dbname=servermail user=usermail password=mailpassword
```

```
default_pass_scheme = SHA512-CRYPT
```

```
password_query = SELECT email as user, password FROM virtual_users WHERE email='%u';
```

cambiamos los permisos para que dovecot sea administrador por el usuario de vmail.

```
chown -R vmail:dovecot /etc/dovecot
```

```
chmod -R o-rwx /etc/dovecot
```

```
##Uncomment inet_listener_imap and modify to port 0
```

```
service imap-login {  
    inet_listener imap {  
        port = 0  
    }  
}
```

```
#Create LMTP socket and this configurations
```

```
service lmtp {  
    unix_listener /var/spool/postfix/private/dovecot-lmtp {  
        mode = 0600  
        user = postfix  
        group = postfix  
    }  
    #inet_listener lmtp {  
        # Avoid making LMTP visible for the entire internet  
        #address =  
        #port =  
    #}
```

```

}
service auth {

    unix_listener /var/spool/postfix/private/auth {
        mode = 0666
        user = postfix
        group = postfix
    }

    unix_listener auth-userdb {
        mode = 0600
        user = vmail
        #group =
    }

    #unix_listener /var/spool/postfix/private/auth {
    # mode = 0666
    #}

    user = dovecot
}

service auth-worker {
    # Auth worker process is run as root by default, so that it can access
    # /etc/shadow. If this isn't necessary, the user should be changed to
    # $default_internal_user.
    user = vmail
}

```

Configuramos el certificado ssl, para dovecot.

```

# nano /etc/dovecot/conf.d/10-ssl.conf

ssl = required

ssl_cert = </etc/ssl/certs/dovecot.pem
ssl_key = </etc/ssl/private/dovecot.pem

service dovecot restart

```

CONCLUSIÓN

Podemos concluir que dentro del trabajo se esta buscando la implementación de un servidor de correo, capaz de dar servicios de creación de usuario y envíos y recepción de correos, por medio de los servicios de SMTP. Este servidor se implementará en un servidor local donde se usará el servicio de DNS para ver la interacción de envío y recepción de correos. El proyecto se realiza con la ayuda de una serie de paquetes de software libre, como lo es POSTFIX, IRedMail entre otros. En el trabajo nos deja claro la importancia de este tipo de servicios en la actualidad, dicho servicio es un generador de comunicación a nivel mundial y conociendo el funcionamiento de este tipo de servidores, nos puede servir para implementar ya sea con fines educativos o laborales. Se habla también sobre la configuración de protocolos para que sea posible este tipo de servicios, y la importancia de estudiar y analizar la forma de como configurar, para tratar de realizar un trabajo completo desde el punto de vista funcional. Otro factor muy importante que se observa es el análisis de su viabilidad y ya que este es un proyecto totalmente educativo.

BIBLIOGRAFÍA

- ⇒ Jorge. (n.d.). Configurar servidor de correo en Debian con postfix y dovecot. Retrieved March 6, 2015, from <http://nksistemas.com/configurar-servidor-de-correo-en-debian-con-postfix-ydovecot/>
- ⇒ Postfix. (2013, October 2). In *Wikipedia, la enciclopedia libre*. Retrieved from <http://es.wikipedia.org/w/index.php?title=Postfix&oldid=69969463>
- ⇒ Servidor HTTP Apache. (2014, September 25). In *Wikipedia, la enciclopedia libre*. Retrieved from http://es.wikipedia.org/w/index.php?title=Servidor_HTTP_Apache&oldid=77175509
- ⇒ Wiki-Geeks: Instalacion/Correo. (n.d.). Retrieved March 6, 2015, from <http://geeks.ues.edu.sv/wiki/index.php?n=Instalacion.Correo>