

INGENIERÍA DE SERVIDORES (2016-2017)
GRADO EN INGENIERÍA INFORMÁTICA
UNIVERSIDAD DE GRANADA

Memoria Práctica 5

Guillermo Montes Martos

20 de enero de 2017

Índice

1. Cuestión 1	4
1.1. Al modificar los valores del kernel mediante el comando <code>sysctl</code> , no logramos que persistan después de reiniciar la máquina. ¿Qué archivo hay que editar para que los cambios sean permanentes?	4
2. Cuestión 2	5
2.1. ¿Con qué opción se muestran todos los parámetros modificables en tiempo de ejecución? Elija dos parámetros y explique, en dos líneas, qué función tienen.	5
3. Cuestión 3	6
3.1. a) Realice una copia de seguridad del registro y restáurela, ilustre el proceso con capturas.	6
3.2. b) Abra una ventana mostrando el editor del registro.	9
4. Cuestión 4	10
4.1. Enumere qué elementos se pueden configurar en Apache y en IIS para que Moodle funcione mejor.	10
5. Cuestión 5	11
5.1. Ajuste la compresión en el servidor y analice su comportamiento usando varios valores para el tamaño de archivo a partir del cual comprimir. Para comprobar que está comprimiendo puede usar el navegador o comandos como <code>curl</code> (see url) o <code>lynx</code> . Muestre capturas de pantalla de todo el proceso.	11
6. Cuestión 6	16
6.1. a) Usted parte de un SO con ciertos parámetros definidos en la instalación (Práctica 1), ya sabe instalar servicios (Práctica 2) y cómo monitorizarlos (Práctica 3) cuando los somete a cargas (Práctica 4). Al igual que ha visto cómo se puede mejorar un servidor web (Práctica 5 Sección 3.1), elija un servicio (el que usted quiera) y modifique un parámetro para mejorar su comportamiento.	16
6.2. b) Monitorice el servicio antes y después de la modificación del parámetro aplicando cargas al sistema (antes y después) mostrando los resultados de la monitorización.	16

Índice de figuras

1.1. Estableciendo el parámetro de prueba de <code>sysctl</code>	4
1.2. Comprobando el parámetro modificado de <code>sysctl</code>	5
2.1. Dos parámetros cualquiera listados con <code>sysctl -a</code>	6
3.1. Abriendo el editor del registro en Windows Server.	6

3.2.	Realizando una copia de seguridad del registro en Windows Server.	7
3.3.	Guardando la copia de seguridad del registro en Windows Server.	7
3.4.	Copia de seguridad del registro en Windows Server en Windows Server. . .	8
3.5.	Restaurando copia de seguridad del registro en Windows Server.	9
3.6.	Editor del registro en Windows Server.	9
5.1.	"Administrador del servidor" en Windows Server.	11
5.2.	Añadiendo la compresión de contenido dinámico a IIS.	12
5.3.	Confirmación anterior a la instalación.	12
5.4.	Menú de Inicio.	13
5.5.	Herramientas administrativas.	13
5.6.	Administrador de IIS.	14
5.7.	Compresión a nivel de servidor en IIS.	14
5.8.	Prueba realizada con curl y la compresión desactivada.	15
5.9.	Prueba realizada con curl y la compresión activada.	15
6.1.	Configuración de DNS.	16
6.2.	Configuración inicial de postfix.	17
6.3.	Dominio con el cual queremos configurar nuestro servidor.	17
6.4.	Creación de tablas en la base de datos.	18
6.5.	Inserción de tuplas en las tablas creadas.	18
6.6.	Configuración principal de postfix.	20
6.7.	Configuración inicial de Postfix.	21
6.8.	Creación y configuración del usuario vmail.	21
6.9.	Configuración de auth-sql.conf.	22
6.10.	Configuración de certificados en 10-ssl.conf.	23
6.11.	Cambio de usuario y de permisos en /etc/dovecot/.	24
6.12.	Parámetros de conexión con Thunderbird	24
6.13.	Fallo encontrado en el log del servidor de correo.	25
6.14.	Estableciendo los certificados en postfix.	25
6.15.	Estableciendo los certificados en dovecot.	25
6.16.	Conexión válida de Thunderbird con el servidor de correo.	26
6.17.	Correos enviados con la cuenta de prueba.	26
6.18.	Correo recibido en cuenta de gmail.	27

1. Cuestión 1

1.1. Al modificar los valores del kernel mediante el comando `sysctl`, no logramos que persistan después de reiniciar la máquina. ¿Qué archivo hay que editar para que los cambios sean permanentes?

Para la realización de este ejercicio, buscamos información al respecto en la documentación oficial de RedHat [1]. En ella encontramos que, como en la gran mayoría de la configuración modificable en Linux, existen al menos dos maneras para alterarla. La primera es modificando un archivo de configuración general de los parámetros del kernel, concretamente `/etc/sysctl.conf`. A este fichero podemos añadir los parámetros que creamos oportunos y estos seguirán vigentes a pesar de reiniciar la máquina. El otro método se trata de añadir sendos archivos de configuración al directorio `/etc/sysctl.d/`, una solución más limpia y adecuada si queremos añadir parámetros al kernel de manera experimental. Una vez modificados los parámetros pertinentes, debemos de actualizar la configuración vigente en la máquina, ejecutando el comando `sysctl -p <fichero.conf>` [2], de manera que se cargara en tiempo de ejecución los parámetros incluidos en el fichero especificado. Si, por el contrario, no se especifica ningún fichero, dichos parámetros se leerán por defecto del archivo `/etc/sysctl.conf`.

Como ejemplo, vamos a modificar un parámetro cualquiera de nuestra máquina virtual con UbuntuServer. Se modificará concretamente el parámetro `net.ipv4.icmp_echo_ignore_all` creando un nuevo archivo de configuración en el directorio `/etc/sysctl.d/`.

```
[03/01/17 gmm@ubuntu-server:~] $ sudo sysctl -a | grep net.ipv4.icmp_echo_ignore_all
net.ipv4.icmp_echo_ignore_all = 1
sysctl: leyendo clave «net.ipv6.conf.all.stable_secret»
sysctl: leyendo clave «net.ipv6.conf.default.stable_secret»
sysctl: leyendo clave «net.ipv6.conf.enp0s3.stable_secret»
sysctl: leyendo clave «net.ipv6.conf.enp0s8.stable_secret»
sysctl: leyendo clave «net.ipv6.conf.lo.stable_secret»
[03/01/17 gmm@ubuntu-server:~] $ sudo sh -c "echo 'net.ipv4.icmp_echo_ignore_all = 0' > /etc/sysctl.d
/prueba.conf"
[03/01/17 gmm@ubuntu-server:~] $ cat /etc/sysctl.d/prueba.conf
net.ipv4.icmp_echo_ignore_all = 0
[03/01/17 gmm@ubuntu-server:~] $ sudo sysctl -p /etc/sysctl.d/prueba.conf
net.ipv4.icmp_echo_ignore_all = 0
[03/01/17 gmm@ubuntu-server:~] $ sudo sysctl -a | grep net.ipv4.icmp_echo_ignore_all
net.ipv4.icmp_echo_ignore_all = 0
sysctl: leyendo clave «net.ipv6.conf.all.stable_secret»
sysctl: leyendo clave «net.ipv6.conf.default.stable_secret»
sysctl: leyendo clave «net.ipv6.conf.enp0s3.stable_secret»
sysctl: leyendo clave «net.ipv6.conf.enp0s8.stable_secret»
sysctl: leyendo clave «net.ipv6.conf.lo.stable_secret»
[03/01/17 gmm@ubuntu-server:~] $ sudo reboot
```

Figura 1.1: Estableciendo el parámetro de prueba de `sysctl`.

Una vez modificado, reiniciamos la máquina y listamos los parámetros que se encuentran vigentes en este momento (mediante el comando `sysctl -a`) y, comprobamos como, efectivamente, el parámetro ha quedado guardado a pesar del reinicio.

```

[03/01/17 gmm@ubuntuserver:~] $ sudo sysctl -a | grep net.ipv4.icmp_echo_ignore_all
[sudo] password for gmm:
net.ipv4.icmp_echo_ignore_all = 0
sysctl: leyendo clave «net.ipv6.conf.all.stable_secret»
sysctl: leyendo clave «net.ipv6.conf.default.stable_secret»
sysctl: leyendo clave «net.ipv6.conf.enp0s3.stable_secret»
sysctl: leyendo clave «net.ipv6.conf.enp0s8.stable_secret»
sysctl: leyendo clave «net.ipv6.conf.lo.stable_secret»
[03/01/17 gmm@ubuntuserver:~] $

```

Figura 1.2: Comprobando el parámetro modificado de sysctl.

2. Cuestión 2

2.1. ¿Con qué opción se muestran todos los parámetros modificables en tiempo de ejecución? Elija dos parámetros y explique, en dos líneas, qué función tienen.

Como ya comprobamos en el anterior ejercicio, la opción del comando *sysctl* para mostrar todos los parámetros modificables en tiempo de ejecución o *runtime* es *-a* o *-A* (ambas tienen el mismo comportamiento [2]).

Tal y como nos piden, listamos los parámetros de nuestra máquina virtual con UbuntuServer (*sysctl -a* / *less* para poder verlos de una manera sencilla) y elegimos dos al azar para explicarlos:

- `net.ipv4.ip_forward`: este parámetro activa el enrutamiento de nuestra máquina Linux. En otras palabras, permite a nuestro PC actuar como puerta de enlace o router [3].
- `kernel.pid_max`: establece el valor máximo de PID (Process ID) a asignar por el kernel [4]. Si se alcanza dicho valor, se ajusta el PID a un valor mínimo menor que el *pid_max*.

```

[03/01/17 gmm@ubuntuuserver:~] $ sudo sysctl -a | grep "ip_forward "
net.ipv4.ip_forward = 0
sysctl: leyendo clave «net.ipv6.conf.all.stable_secret»
sysctl: leyendo clave «net.ipv6.conf.default.stable_secret»
sysctl: leyendo clave «net.ipv6.conf.enp0s3.stable_secret»
sysctl: leyendo clave «net.ipv6.conf.enp0s8.stable_secret»
sysctl: leyendo clave «net.ipv6.conf.lo.stable_secret»
[03/01/17 gmm@ubuntuuserver:~] $ sudo sysctl -a | grep pid_max
kernel.pid_max = 32768
sysctl: leyendo clave «net.ipv6.conf.all.stable_secret»
sysctl: leyendo clave «net.ipv6.conf.default.stable_secret»
sysctl: leyendo clave «net.ipv6.conf.enp0s3.stable_secret»
sysctl: leyendo clave «net.ipv6.conf.enp0s8.stable_secret»
sysctl: leyendo clave «net.ipv6.conf.lo.stable_secret»
[03/01/17 gmm@ubuntuuserver:~] $ _

```

Figura 2.1: Dos parámetros cualquiera listados con *sysctl -a*.

3. Cuestión 3

3.1. a) Realice una copia de seguridad del registro y restáurela, ilustre el proceso con capturas.

Como disponemos de GUI en nuestro servidor Windows, podemos seguir los mismos pasos necesarios para la realización de una copia de seguridad y restauración del registro explicados por Microsoft en su documentación para SO de escritorio [5].

Lo primero será ejecutar *regedit* con permisos de administrador para abrir el editor del registro.

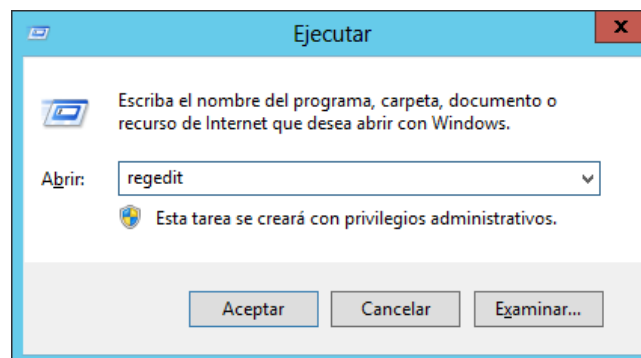


Figura 3.1: Abriendo el editor del registro en Windows Server.

Para realizar la copia de seguridad, tendremos que pinchar para en "Archivo" → "Exportar".

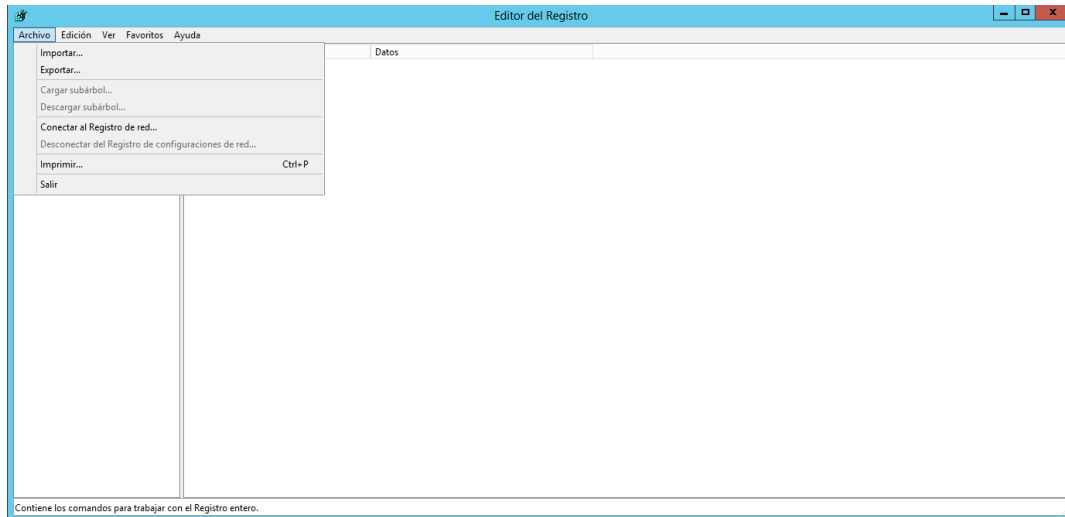


Figura 3.2: Realizando una copia de seguridad del registro en Windows Server.

A continuación, le damos un nombre y la guardamos en la ubicación que deseemos. En nuestro caso, la guardaremos en el *Escritorio* con el nombre *ejercicio3.reg*.

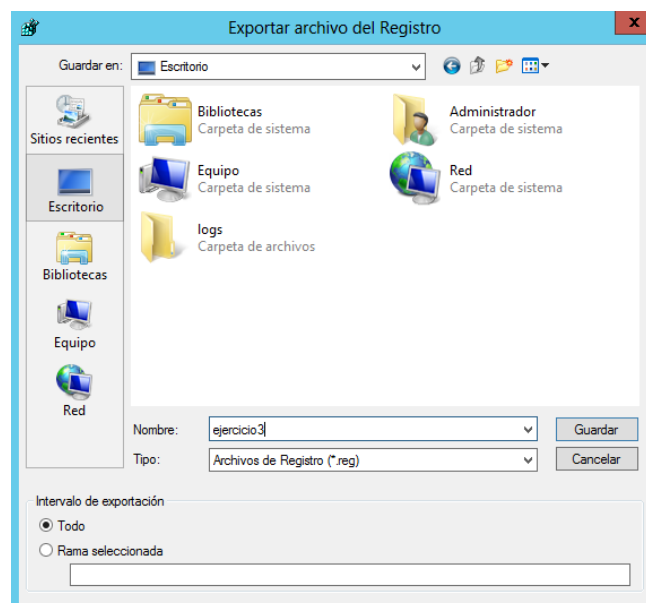


Figura 3.3: Guardando la copia de seguridad del registro en Windows Server.

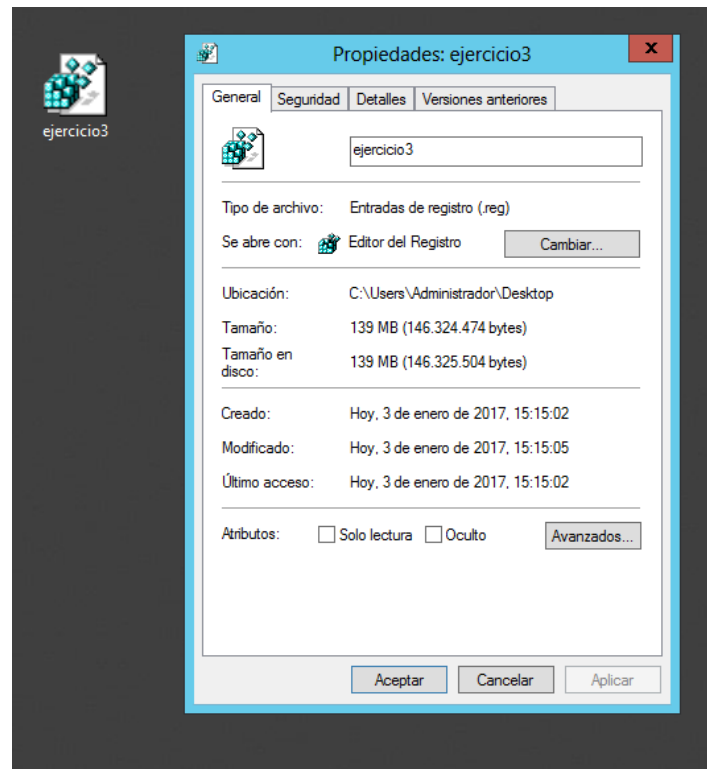


Figura 3.4: Copia de seguridad del registro en Windows Server en Windows Server.

Para restaurar la copia de seguridad realizada, tendremos que volver a abrir el editor del registro, hacer click en "Archivo" → "Importar" y seleccionar el archivo previamente creado.

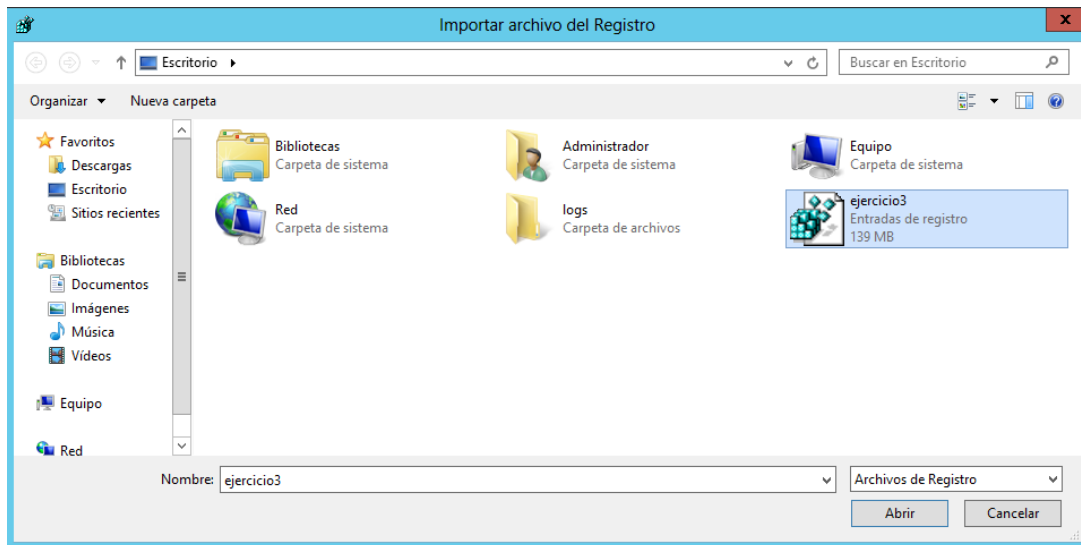


Figura 3.5: Restaurando copia de seguridad del registro en Windows Server.

3.2. b) Abra una ventana mostrando el editor del registro.

Tal y como se explicó en el anterior apartado, para abrir el editor del registro tan solo tendremos que ejecutar el comando *regedit* con permisos de administración.

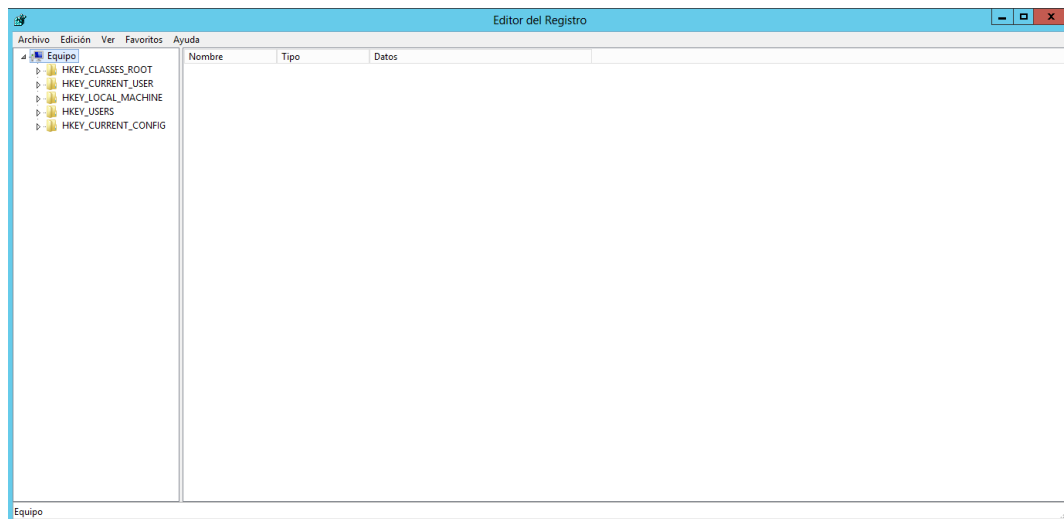


Figura 3.6: Editor del registro en Windows Server.

4. Cuestión 4

4.1. Enumere qué elementos se pueden configurar en Apache y en IIS para que Moodle funcione mejor.

Para consultar dichos elementos de configuración se accede a la página dada en el guión de prácticas [6], donde podemos ver varios apartados, entre ellos la mejora del rendimiento de moodle según el servidor web instalado.

En primer lugar, accedemos a las mejoras indicadas para Apache [7]. En dicha página, se recomiendan bastantes para aumentar el rendimiento de Moodle, entre las cuales destacan las siguientes:

- Establecer correctamente el número máximo de clientes *MaxClients*, para lo cual se propone una fórmula donde se usa el 80 % de la memoria disponible.
- Reducir lo máximo posible el número de módulos de Apache.
- Utilizar la última versión de Apache2.
- En sistemas Linux, establecer el número máximo de peticiones por hijo *MaxRequestsPerChild* como muy bajo entre 20 y 30.
- Fijar el valor del parámetro *KeepAliveTimeout* entre 2 y 5 como mucho.
- Si no se usa ningún fichero *.htaccess*, establecer la variable *AllowOverride* a *None* para evitar su búsqueda.

A continuación, procedemos a la mejoras recomendadas para ISS [8]. Todas las mejoras que podemos realizar se harán en la dirección del registro `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Inetinfo\Parameters\`.

- Establecer *ListenBackLog* entre 2 y 5. Este es el parámetro similar a *KeepAliveTimeout* usado en Apache.
- Ajustar los valores de los parámetros *MemCacheSize* y *MaxCachedFileSize*.
- Crear un DWORD (Double Word o número de 32 bits [9]) llamado *ObjectCacheTTL* para establecer el tiempo que un objeto caché permanece en memoria.

5. Cuestión 5

5.1. Ajuste la compresión en el servidor y analice su comportamiento usando varios valores para el tamaño de archivo a partir del cual comprimir. Para comprobar que está comprimiendo puede usar el navegador o comandos como curl (see url) o lynx. Muestre capturas de pantalla de todo el proceso.

Lo primero que tendremos que hacer es instalar esta característica, puesto que en la instalación que se realizó previamente no se tuvo en cuenta. Para ello, abrimos el "Administrador del servidor" y hacemos click en "Agregar roles y características".

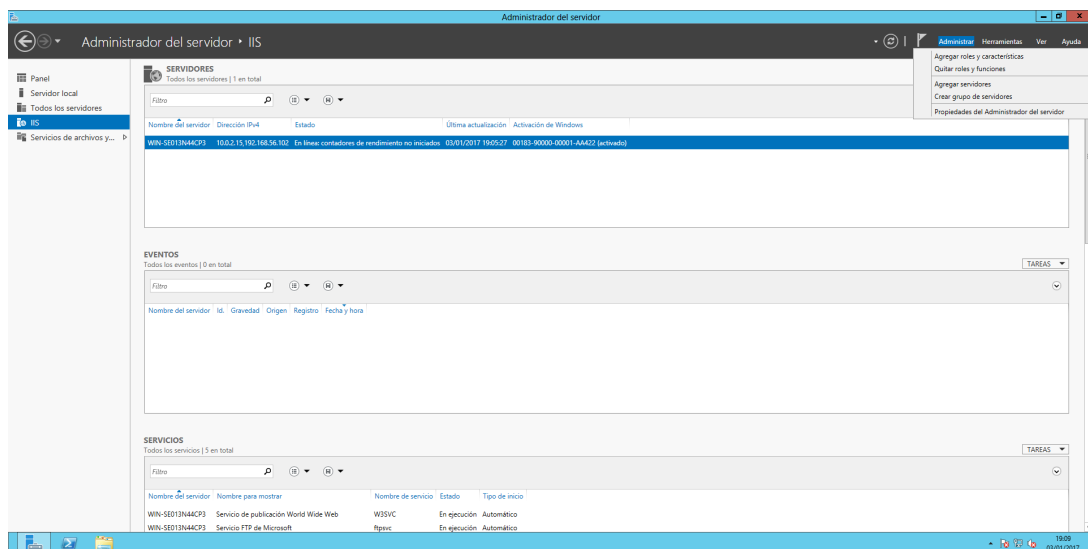


Figura 5.1: "Administrador del servidor" en Windows Server.

Se nos mostrará un diálogo, donde tendremos que seleccionar nuestro servidor y elegir la característica a añadir, en este caso "Compresión de contenido dinámico".

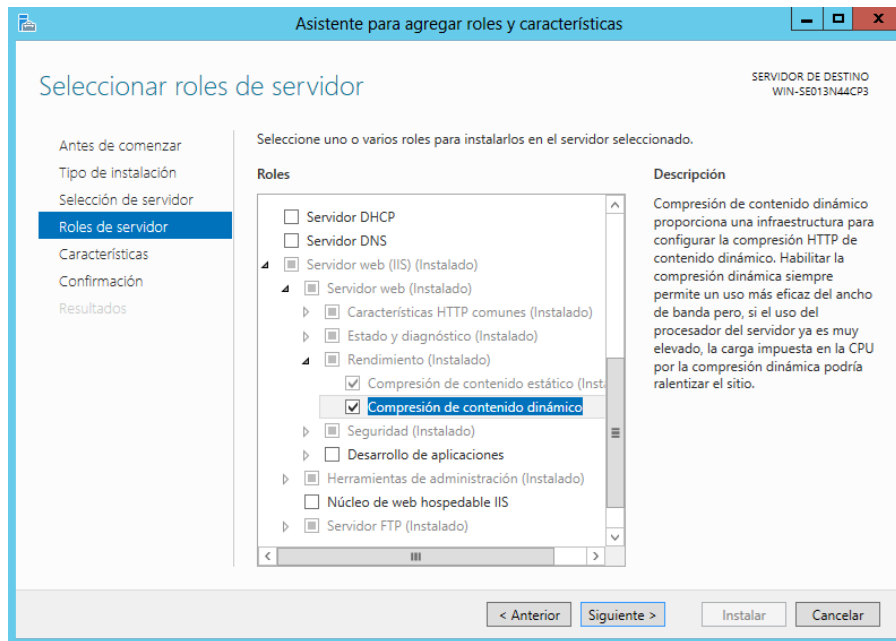


Figura 5.2: Añadiendo la compresión de contenido dinámico a IIS.

Aceptamos el diálogo de confirmación y esperamos a que termine la instalación.

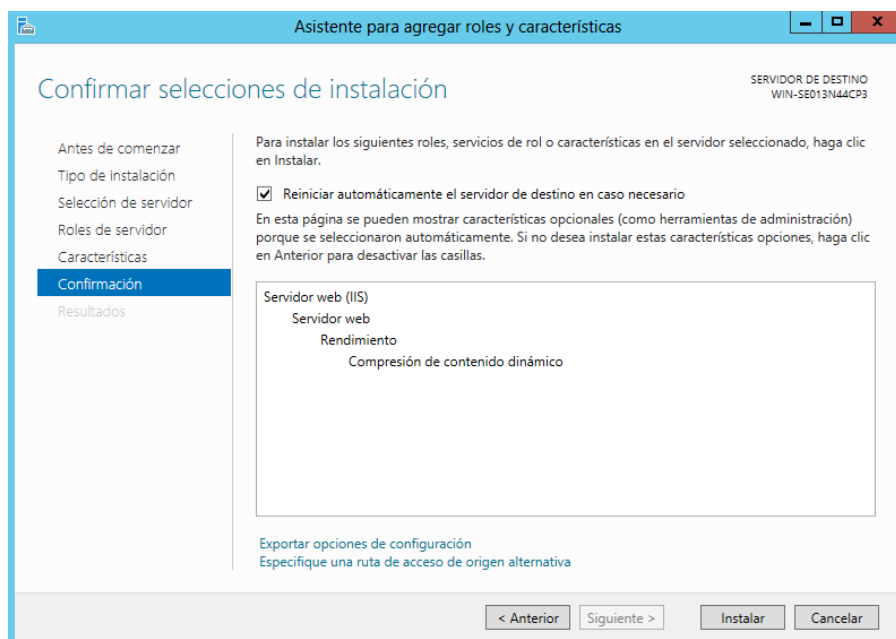


Figura 5.3: Confirmación anterior a la instalación.

Una vez instalado, procedemos a activarlo [10]. Abrimos el menú de Inicio y pinchamos

en "Herramientas administrativas".

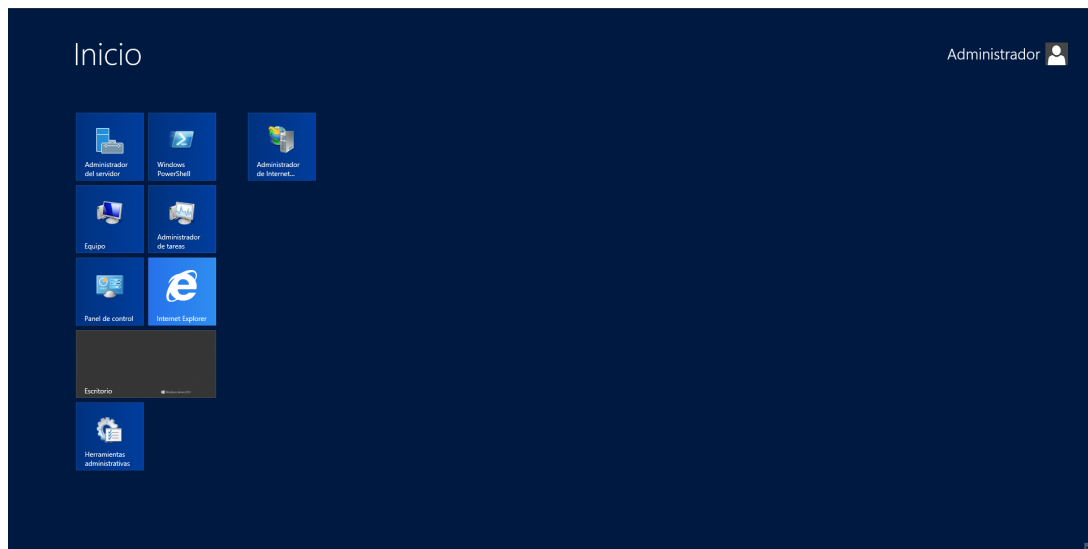


Figura 5.4: Menú de Inicio.

Dentro, abrimos el "Administrador de Internet Information Services (IIS)".

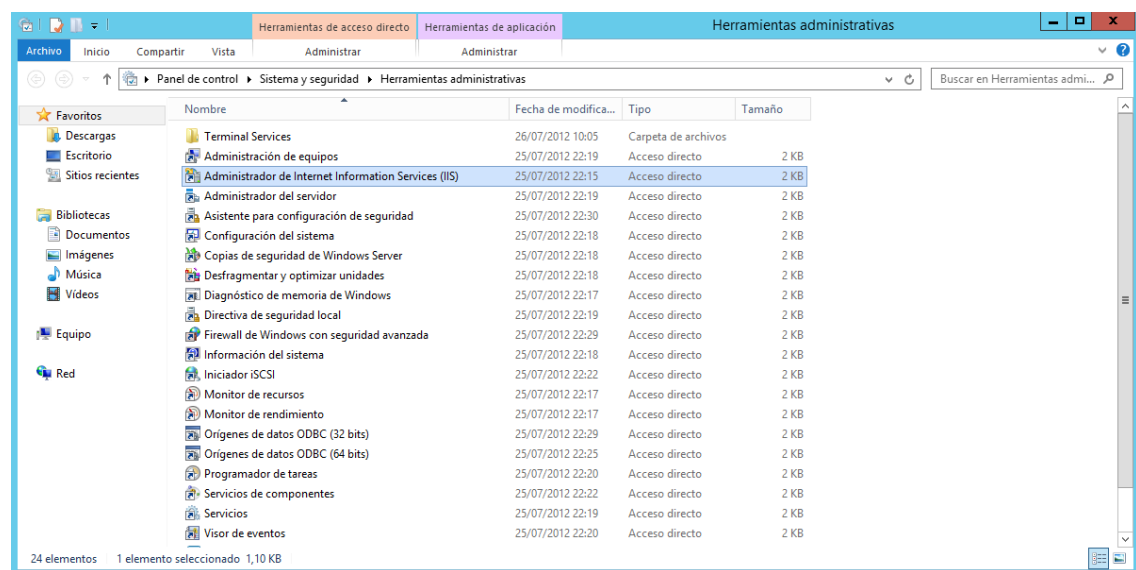


Figura 5.5: Herramientas administrativas.

Seleccionamos el servidor y pinchamos en "Compresión", dentro del submenú "IIS".

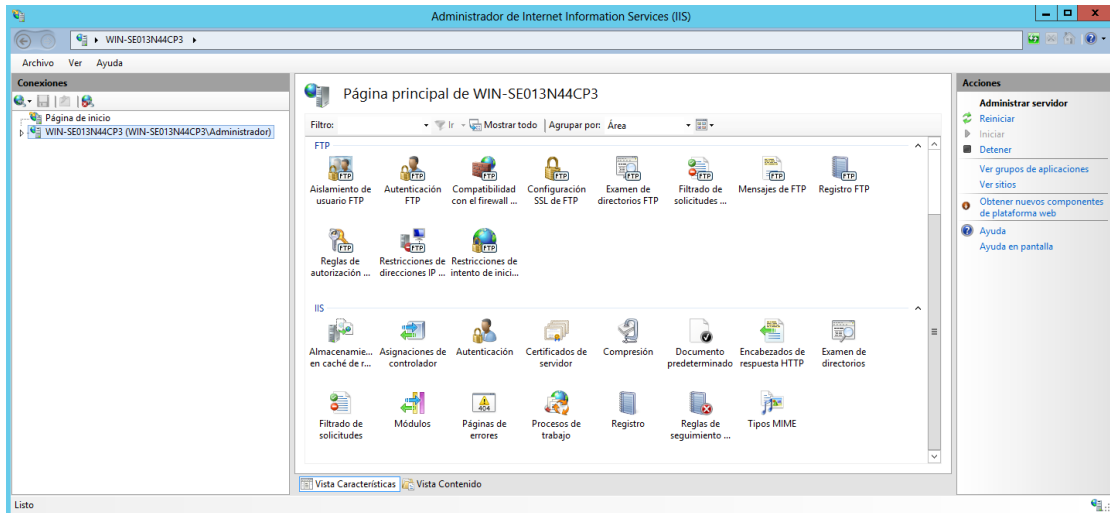


Figura 5.6: Administrador de IIS.

Una vez en el menú indicado, activamos todos los *checkboxes* y asignamos un valor de bytes a partir del cual el servidor comenzará a comprimir los archivos.

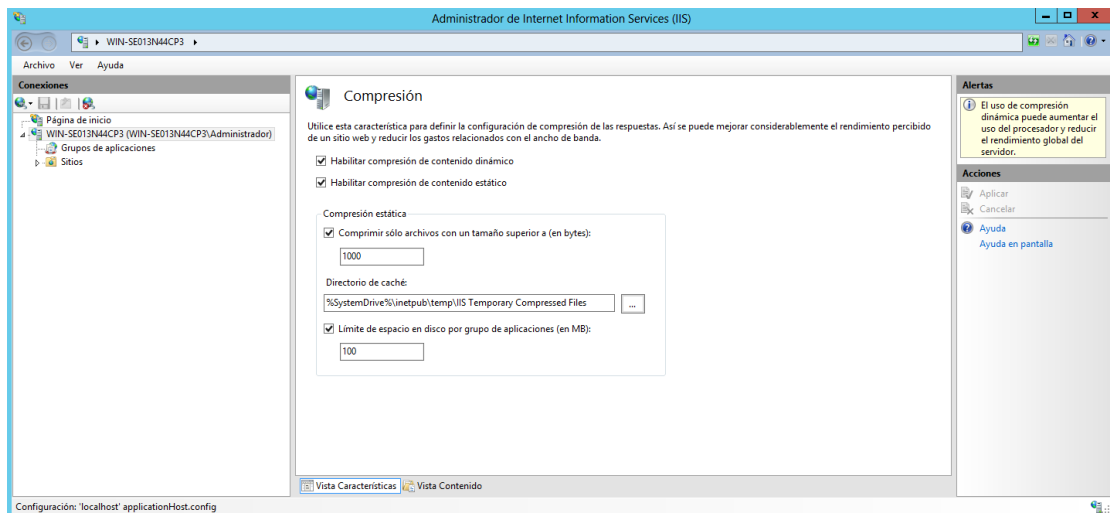


Figura 5.7: Compresión a nivel de servidor en IIS.

Para comprobar que la compresión funciona correctamente, usaremos el comando `curl` [11] (una herramienta para transferir información desde o a un servidor bajo múltiples protocolos) con las opciones `-H 'Accept-Encoding: gzip, deflate'` y `-I`. La primera especifica la cabecera adicional a usar para realizar la petición HTTP. Esta es necesaria para poder aceptar contenido con compresión. El segundo parámetro sirve para obtener la cabecera del fichero HTTP únicamente. De esta manera, obtenemos una salida limpia. Así, se

procederá a realizar sendas pruebas con el comando curl activando y desactivando la compresión en el servidor IIS, comprobando como, efectivamente, el tamaño ("Content-Length") disminuye con la compresión activada.

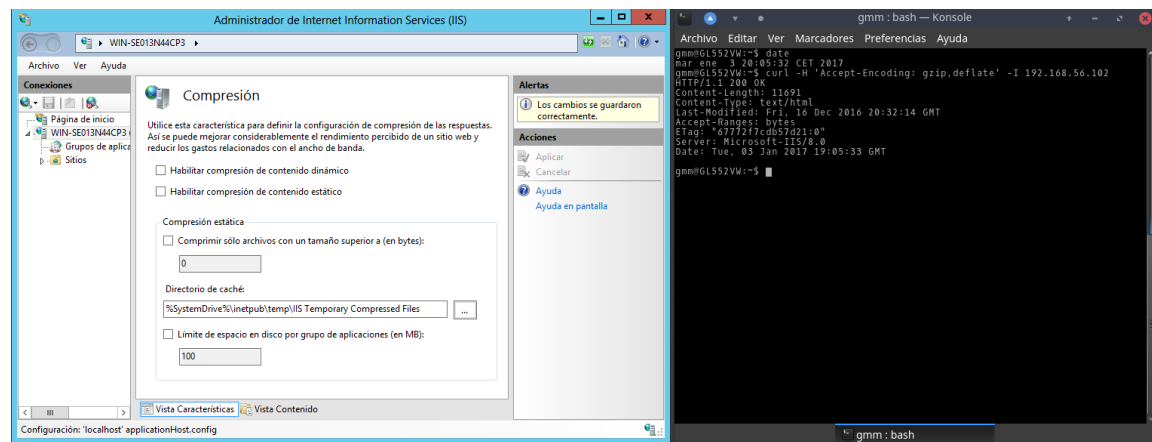


Figura 5.8: Prueba realizada con curl y la compresión desactivada.

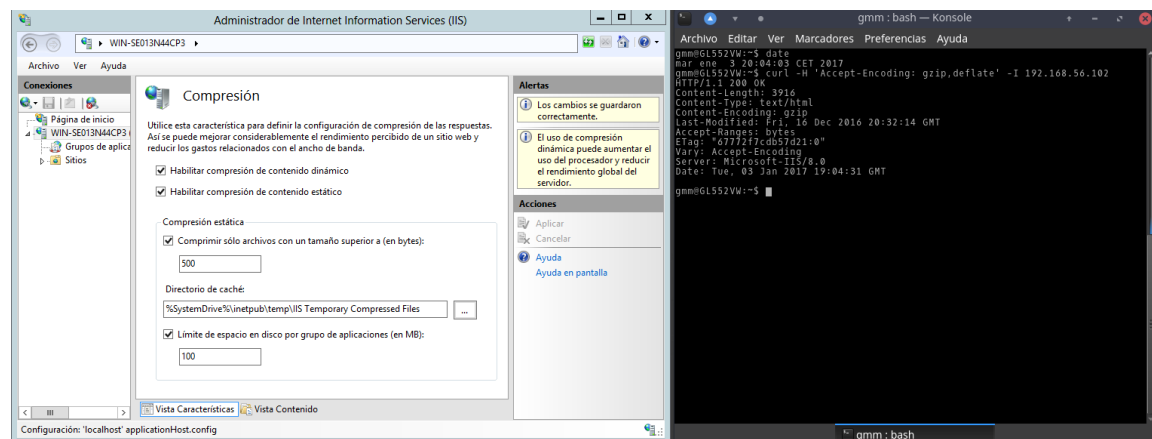


Figura 5.9: Prueba realizada con curl y la compresión activada.

6. Cuestión 6

- 6.1. a) Usted parte de un SO con ciertos parámetros definidos en la instalación (Práctica 1), ya sabe instalar servicios (Práctica 2) y cómo monitorizarlos (Práctica 3) cuando los somete a cargas (Práctica 4). Al igual que ha visto cómo se puede mejorar un servidor web (Práctica 5 Sección 3.1), elija un servicio (el que usted quiera) y modifique un parámetro para mejorar su comportamiento.
- 6.2. b) Monitorice el servicio antes y después de la modificación del parámetro aplicando cargas al sistema (antes y después) mostrando los resultados de la monitorización.

Para la realización de este ejercicio, se ha escogido la instalación de un servidor de correo, la cual tendrá lugar en un VPS (Virtual Private Server) real de la compañía ArubaCloud [12], no en una máquina virtual. Este, que cuenta con UbuntuServer 16.04, fue adquirido con anterioridad, al igual que el dominio con el cual se implementará el servidor de correo, uno gratuito con terminación .tk perteneciente al país de Tokelau. Solo se explicará el apartado de instalación, ya que, tras consultar al profesor, valoró de forma positiva todo el trabajo que esto conlleva.

El primer paso consistiría en configurar correctamente los DNS (Domain Name Server) para poder redireccionar correctamente a nuestro VPS usando el dominio adquirido.

Name	Type	TTL	Target	
	A	14400	89.36.222.93	Delete
CORREO	A	14400	89.36.222.93	Delete
WWW	A	300	89.36.222.93	Delete
	MX	14400	correo.gulleando.tk	Delete
			Priority:	Delete

Figura 6.1: Configuración de DNS.

Aunque las anteriores prácticas se realizaron en una máquina virtual, la instalación de LAMP ya tuvo lugar en este VPS, por lo cual no se considera necesaria su explicación. Procedemos entonces con la instalación de los servicios necesarios, para lo cual se seguirá un tutorial ofrecido por DigitalOcean [13]. El segundo paso será instalar los programas requeridos, postfix y dovecot (y una serie de módulos complementarios), para lo cual ejecutamos *sudo apt-get install postfix postfix-mysql dovecot-core dovecot-imapd dovecot-lmtpd dovecot-mysql*.

Postfix [14] es un servidor de transferencia de correo. En otras palabras, es el encargado del envío de los correos electrónicos. En su instalación, nos aparecerá un terminal interactivo donde tendremos que seleccionar "Internet Site" e introducir el dominio con

el cual queremos configurarlo.

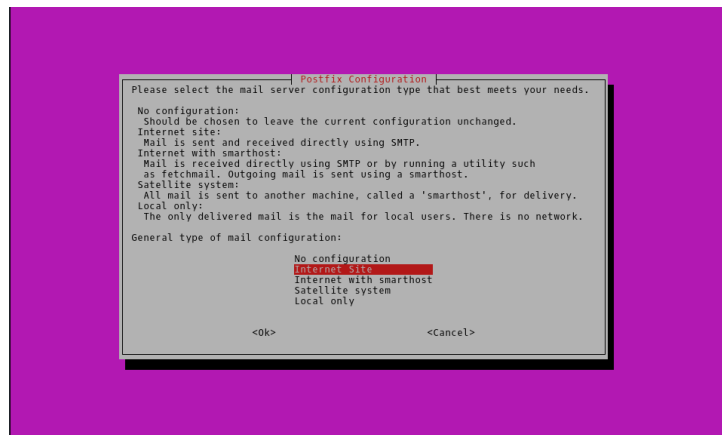


Figura 6.2: Configuración inicial de postfix.

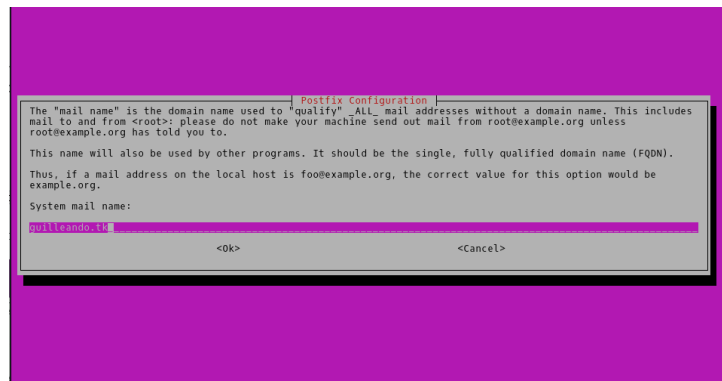


Figura 6.3: Dominio con el cual queremos configurar nuestro servidor.

Hecho esto, tendremos que configurar la base de datos necesaria para almacenar los usuarios, los dominios, etc. Para eso, creamos una nueva base de datos con el comando `mysqladmin -p create <nombre BBDD>` y ejecutamos mysql con el usuario administrador. Dentro de la consola de mysql, tendremos dar permisos a un usuario para manejar dicha BBDD y recargar los permisos. Como ejemplos, en la realización se usó "servidor-Correo" para la BBDD y "adminCorreo" para el nombre.

Listing 1: Estableciendo privilegios

```
GRANT SELECT ON servidorCorreo.* TO 'adminCorreo'@'127.0.0.1' IDENTIFIED BY '
    mailpassword';
FLUSH PRIVILEGES;
```

A continuación, nos situamos en la base de datos y creamos las tablas necesarias. Aunque en las capturas y en el tutorial aparece la creación de una tabla de "alias, se omitirá dada su poca relevancia. También insertamos unas tuplas de prueba en las tablas creadas.

```
mysql>
mysql> use servidorCorreo;
Database changed
mysql> create table `virtual_domains` (`id` INT NOT NULL AUTO_INCREMENT, `name` VARCHAR(50) NOT NULL, PRIMARY KEY (`id`
`) ) ENGINE=InnoDB DEFAULT CHARSET=utf8;
Query OK, 0 rows affected (0.02 sec)

mysql> create table `virtual_users` (`id` INT NOT NULL AUTO_INCREMENT, `domain_id` INT NOT NULL, `password` VARCHAR(10
6) NOT NULL, `email` VARCHAR(120) NOT NULL, PRIMARY KEY(`id`), UNIQUE KEY `email` (`email`), FOREIGN KEY(domain_id) RE
FERENCES virtual_domains(id) ON DELETE CASCADE ) ENGINE=InnoDB DEFAULT CHARSET=utf8;
Query OK, 0 rows affected (0.02 sec)
```

Figura 6.4: Creación de tablas en la base de datos.

```
mysql> create table `virtual_aliases` (`id` INT NOT NULL AUTO_INCREMENT, `domain_id` INT NOT NULL, `source` VARCHAR(10
0) 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;
Query OK, 0 rows affected (0.01 sec)

mysql> INSERT INTO `servidorCorreo`.`virtual_domains`(`id`, `name`) VALUES('1', 'correo.guilleando.tk');
Query OK, 1 row affected (0.01 sec)

mysql> INSERT INTO `servidorCorreo`.`virtual_users`(`id`,`domain_id`,`password`,`email`) VALUES('1', '1', ENCRYPT('con
traseña', CONCAT('$6$', SUBSTRING(SHA(RAND()), -16))), 'prueba@guilleando.tk');
ERROR 1048 (23000): Column 'password' cannot be null
mysql> INSERT INTO `servidorCorreo`.`virtual_users`(`id`,`domain_id`,`password`,`email`) VALUES('1', '1', ENCRYPT('con
traseña', CONCAT('$6$', SUBSTRING(SHA(RAND()), -16))), 'prueba@guilleando.tk');
Query OK, 1 row affected, 1 warning (0.01 sec)

mysql> INSERT INTO `servidorCorreo`.`virtual_aliases`(`id`,`domain_id`,`source`,`destination`) VALUES('1','1','alias@g
uilleando.tk','prueba@guilleando.tk');
Query OK, 1 row affected (0.00 sec)

mysql>
```

Figura 6.5: Inserción de tuplas en las tablas creadas.

Una vez realizado, pasamos a la configuración de postfix. Hacemos un backup del archivo principal de configuración (/etc/postfix/main.cf) e introducimos lo siguiente (como aclaración, no se han podido incluir tildes en los comentarios por problemas con latex).

Listing 2: /etc/postfix/main.cf

```
# Parametros iniciales
smtpd_banner = $myhostname ESMTP $mail_name (Ubuntu)
biff = no
append_dot_mydomain = no
readme_directory = /usr/share/doc/postfix

# Parametros TLS
smtpd_tls_cert_file=/etc/ssl/certs/ssl-cert-snakeoil.pem
smtpd_tls_key_file=/etc/ssl/certs/ssl-cert-snakeoil.key
smtpd_use_tls=yes
```

```

smtpd_tls_auth_only = yes
smtpd_tls_CAfile = /etc/postfix/cacert.pem
smtp_tls_CAfile = /etc/postfix/cacert.pem
smtpd_sasl_type = dovecot
smtpd_sasl_path = private/auth
smtpd_sasl_auth_enable = yes
smtpd_recipient_restrictions = permit_sasl_authenticated,permit_mynetworks,
    reject_unauth_destination
smtp_sasl_security_options = noanonymous

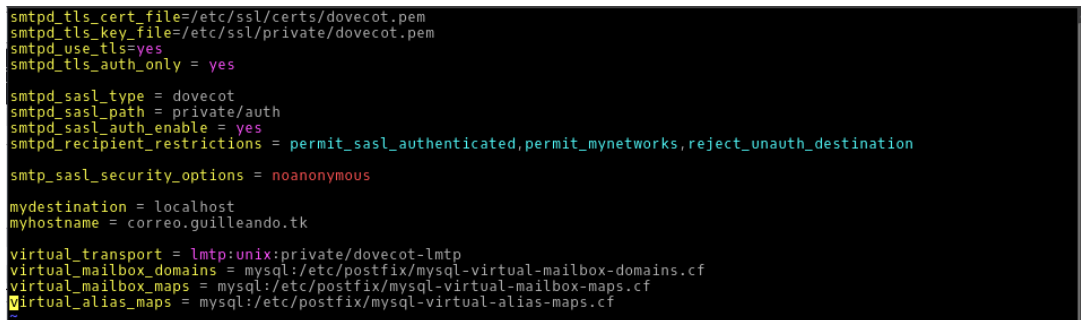
# Desactivar Poodle
smtp_tls_security_level = may
smtpd_tls_mandatory_protocols=!SSLv2,!SSLv3
smtp_tls_mandatory_protocols=!SSLv2,!SSLv3
smtpd_tls_protocols=!SSLv2,!SSLv3
smtp_tls_protocols=!SSLv2,!SSLv3

# Lista de cifrados SSL
tls_preempt_cipherlist = yes
smtpd_tls_mandatory_ciphers = high
tls_high_cipherlist = ECDHE-RSA-AES256-GCM-SHA384:ECDHE-ECDSA-AES256-GCM-
    SHA384:ECDHE-RSA-AES256-SHA384:ECDHE-ECDSA-AES256-SHA384:DHE-DSS-AES256-
    GCM-SHA384:DHE-RSA-AES256-GCM-SHA384:DHE-RSA-AES256-SHA256:DHE-DSS-AES256-
    -SHA256:ADH-AES256-GCM-SHA384:ADH-AES256-SHA256:ECDH-RSA-AES256-GCM-
    SHA384:ECDH-ECDSA-AES256-GCM-SHA384:ECDH-RSA-AES256-SHA384:ECDH-ECDSA-
    AES256-SHA384:AES256-GCM-SHA384:AES256-SHA256:ECDHE-RSA-AES128-GCM-SHA256
    :ECDHE-ECDSA-AES128-GCM-SHA256:ECDHE-RSA-AES128-SHA256:ECDHE-ECDSA-AES128
    -SHA256:DHE-DSS-AES128-GCM-SHA256:DHE-RSA-AES128-GCM-SHA256:DHE-RSA-
    AES128-SHA256:DHE-DSS-AES128-SHA256:ADH-AES128-GCM-SHA256:ADH-AES128-
    SHA256:ECDH-RSA-AES128-GCM-SHA256:ECDH-ECDSA-AES128-GCM-SHA256:ECDH-RSA-
    AES128-SHA256:ECDH-ECDSA-AES128-SHA256:AES128-GCM-SHA256:AES128-SHA256:
    NULL-SHA256

# Dominios
mydestination = localhost, localhost.localdomain
myhostname = correo.guilleando.tk

# Activacion de dominios virtuales y localizacion
virtual_transport = lmtp:unix:private/dovecot-lmtp
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
virtual_uid_maps = static:5000
virtual_gid_maps = static:5000

```



```
smtpd_tls_cert_file=/etc/ssl/certs/dovecot.pem
smtpd_tls_key_file=/etc/ssl/private/dovecot.pem
smtpd_use_tls=yes
smtpd_tls_auth_only = yes

smtpd_sasl_type = dovecot
smtpd_sasl_path = private/auth
smtpd_sasl_auth_enable = yes
smtpd_recipient_restrictions = permit_sasl_authenticated,permit_mynetworks,reject_unauth_destination

smtp_sasl_security_options = noanonymous

mydestination = localhost
myhostname = correo.guilleando.tk

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

Figura 6.6: Configuración principal de postfix.

Los parámetros iniciales son informativos y vienen por defecto. Los parámetros TLS nos permiten el uso de este protocolo en nuestro servidor de correo. A continuación desactivamos Poodle [15], una vulnerabilidad encontrada en SSL, de manera que mantengamos nuestro servidor lo más seguro posible. Establecemos la lista de cifrados para SSL y los dominios sobre los que trabajará postfix. Por último, activamos el uso de dominios virtuales, para lo cual tendremos que crear una serie de ficheros gracias a los cuales definimos las conexiones con la base de datos.

Listing 3: /etc/postfix/mysql-virtual-mailbox-domains.cf

```
user = adminCorreo
password = mailpassword
hosts = 127.0.0.1
dbname = servidorCorreo
query = SELECT 1 FROM virtual_domains WHERE name='%s'
```

Listing 4: /etc/postfix/mysql-virtual-mailbox-domains.cf

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

Por último, debemos descomentar unas líneas del fichero /etc/postfix/master.cf para hacer uso de TLS.

```

submission inet n      -       -       -       smtpd
-o syslog_name=postfix/submission
-o smtpd_tls_security_level=encrypt
-o smtpd_sasl_auth_enable=yes
# -o smtpd_reject_unlisted_recipient=no
-o smtpd_client_restrictions=$mua_client_restrictions
-o smtpd_client_restrictions=permit_sasl_authenticated,reject

```

Figura 6.7: Configuración inicial de Postfix.

Dovecot [16] es un servidor de correo que proporciona acceso a protocolos IMAP, POP3, etc. En otras palabras, es el responsable de la gestión de cuentas y carpetas de correo. Es por ello por lo que es necesaria su instalación junto a postfix. Procedemos ahora con su modificación. Primero tratamos el archivo de configuración principal de dovecot (/etc/dovecot/dovecot.conf), donde tendremos que incluir la línea *protocols = imap lmtp* justo debajo de *!include_try /usr/share/dovecot/protocols.d/*.protocol*. Esto activará los protocolos establecidos. A continuación, tendremos que modificar una serie de ficheros del directorio donde se encuentran los archivos de configuración secundarios (/etc/dovecot/conf.d). El primero será 10-mail.conf, donde tendremos que encontrar las siguientes líneas, descomentarlas y darle los siguientes valores para establecer la ubicación y la propiedad de los ficheros donde se ubicarán los correos.

Listing 5: /etc/dovecot/conf.d/10-mail.conf

```

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

```

Hecho esto, tendremos que crear el usuario y el grupo "vmail", la carpeta donde se guardarán los correos y cambiar su propiedad al usuario y grupo creados.

```

root@UbuntuServer:/home/gmm# vim /etc/dovecot/conf.d/10-mail.conf
root@UbuntuServer:/home/gmm# mkdir -p /var/mail/vhosts/guilleando.tk
root@UbuntuServer:/home/gmm# groupadd -g 5000 vmail
root@UbuntuServer:/home/gmm# useradd -g vmail -u 5000 vmail -d /var/mail
root@UbuntuServer:/home/gmm# chown -R vmail:vmail /var/mail
root@UbuntuServer:/home/gmm# █

```

Figura 6.8: Creación y configuración del usuario vmail.

Seguimos ahora con el fichero /etc/dovecot/conf.d/10-auth.conf, donde tendremos que descomentar la línea que contiene *disable_plaintext_auth = yes* para no permitir el uso de texto plano, modificar la línea *auth_mechanisms* con los parámetros *plain login*, comentar la línea *!include auth-system.conf.ext* y descomentar *!include auth-sql.conf.ext* para habilitar el uso de la base de datos.

En el fichero /etc/dovecot/conf.d/auth-sql.conf.ext tendremos que añadir la línea *args = uid=vmail gid=vmail home=/var/mail/vhosts/%d/%n* en el apartado *userdb* para establecer la localización de los correos.

```

# Path for SQL configuration file, see example-config/dovecot-sql.conf.ext
args = /etc/dovecot/dovecot-sql.conf.ext
}

# "prefetch" user database means that the passdb already provided the
# needed information and there's no need to do a separate userdb lookup.
# <doc/wiki/UserDatabase.Prefetch.txt>
#userdb {
#  driver = prefetch
#}

#userdb {
#  driver = sql
#  args = /etc/dovecot/dovecot-sql.conf.ext
#}

# If you don't have any user-specific settings, you can avoid the user_query
# by using userdb static instead of userdb sql, for example:
# <doc/wiki/UserDatabase.Static.txt>
userdb {
  driver = static
  args = uid=vmail gid=vmail home=/var/mail/vhosts/%d/%n
}
:wq

```

Figura 6.9: Configuración de auth-sql.conf.

Pasamos al archivo `/etc/dovecot/dovecot-sql.conf.ext`, donde tendremos que establecer como driver *mysql*, modificar los parámetros de conexión a la base de datos con la línea *connect = host=127.0.0.1 dbname=servidorCorreo user=adminCorreo password=mailpassword*, la encriptación de la contraseña a *default_pass_scheme = SHA512-CRYPT* y la consulta a la hora de hacer login a *password_query = SELECT email as user, password FROM virtual_users WHERE email='%u'*;

El siguiente fichero es `/etc/dovecot/conf.d/10-master.conf`, donde tendremos que modificar lo siguiente para garantizar la autenticación y el acceso a los correos.

Listing 6: `/etc/dovecot/conf.d/10-master.conf`

```

# Descomentar el puerto y establecerlo a 0
service imap-login {
  inet_listener imap {
    port = 0
  }
}

...

# Crear el socket LMTP con la siguiente configuracion
service lmtp {
  unix_listener /var/spool/postfix/private/dovecot-lmtp {
    mode = 0600
    user = postfix
    group = postfix
  }
}

...

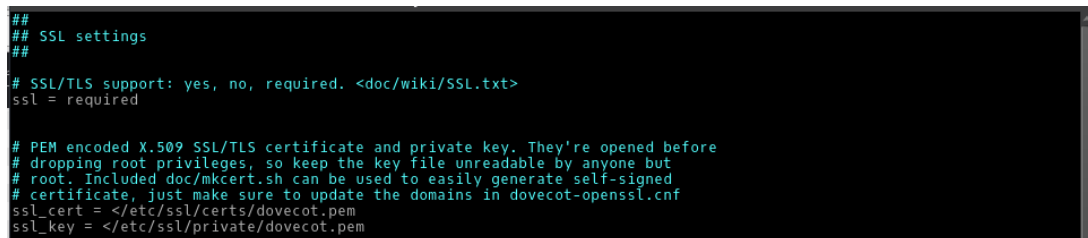
```

```
# Modificar el servicio de autentificacion
service auth {
    unix_listener /var/spool/postfix/private/auth {
        mode = 0666
        user = postfix
        group = postfix
    }
    unix_listener auth-userdb {
        mode = 0600
        user = vmail
        #group =
    }
    user = dovecot
}

...

service auth-worker {
    user = vmail
}
```

En el fichero `etc/dovecot/conf.d/10-ssl.conf` tendremos que establecer los certificados de seguridad requeridos por SSL.



```
##
## SSL settings
##
# SSL/TLS support: yes, no, required. <doc/wiki/SSL.txt>
ssl = required

# PEM encoded X.509 SSL/TLS certificate and private key. They're opened before
# dropping root privileges, so keep the key file unreadable by anyone but
# root. Included doc/mkcert.sh can be used to easily generate self-signed
# certificate, just make sure to update the domains in dovecot-openssl.cnf
ssl_cert = </etc/ssl/certs/dovecot.pem
ssl_key = </etc/ssl/private/dovecot.pem
```

Figura 6.10: Configuración de certificados en 10-ssl.conf.

Hecho esto, cambiamos el usuario y los permisos del directorio `/etc/dovecot` para solo permitir modificaciones por parte del usuario `vmail`.

```

root@UbuntuServer:/home/gmm# vim /etc/dovecot/dovecot.conf
root@UbuntuServer:/home/gmm# vim /etc/dovecot/conf.d/10-mail.conf
root@UbuntuServer:/home/gmm# mkdir -p /var/mail/vhosts/guilleando.tk
root@UbuntuServer:/home/gmm# groupadd -g 5000 vmail
root@UbuntuServer:/home/gmm# useradd -g vmail -u 5000 vmail -d /var/mail
root@UbuntuServer:/home/gmm# chown -R vmail:vmail /var/mail
root@UbuntuServer:/home/gmm# vim /etc/dovecot/conf.d/10-auth.conf
root@UbuntuServer:/home/gmm# vim /etc/dovecot/conf.d/auth-sql.conf.ext
root@UbuntuServer:/home/gmm# vim /etc/dovecot/conf.d/auth-sql.conf.ext
root@UbuntuServer:/home/gmm# vim /etc/dovecot/conf.d/auth-sql.conf.ext
root@UbuntuServer:/home/gmm# vim /etc/dovecot/dovecot-sql.conf.ext
root@UbuntuServer:/home/gmm# chown -R vmail:dovecot /etc/dovecot/
root@UbuntuServer:/home/gmm# chmod -R o-rwx /etc/dovecot/
root@UbuntuServer:/home/gmm# ls -l /etc/dovecot/
total 52
drwxr-x--- 2 vmail dovecot 4096 Jan 17 22:51 conf.d
-rw-r----- 1 vmail dovecot 4478 Jan 17 22:41 dovecot.conf
-rw-r----- 1 vmail dovecot 4401 Jan 17 22:21 dovecot.conf.bak
-rw-r----- 1 vmail dovecot 1507 Mar 16 2016 dovecot-dict-auth.conf.ext
-rw-r----- 1 vmail dovecot 852 Mar 16 2016 dovecot-dict-sql.conf.ext
-rw-r----- 1 vmail dovecot 5681 Jan 17 22:55 dovecot-sql.conf.ext
-rw-r----- 1 vmail dovecot 5612 Jan 17 22:22 dovecot-sql.conf.ext.bak
drwxr-x--- 2 vmail dovecot 4096 Nov 10 15:01 private
-rw-r----- 1 vmail dovecot 121 Nov 10 15:02 README
root@UbuntuServer:/home/gmm# vim /etc/dovecot/conf.d/10-master.conf
root@UbuntuServer:/home/gmm# vim /etc/dovecot/conf.d/10-ssl.conf

```

Figura 6.11: Cambio de usuario y de permisos en /etc/dovecot/.

Según el tutorial de DigitalOcean, ya podríamos reiniciar los servicios postfix y dovecot (*sudo systemctl restart postfix dovecot*) y usar nuestro servidor haciendo login desde un cliente mail en un PC cualquiera. Así, abrimos Thunderbird en nuestro PC y establecemos los parámetros de conexión, pero encontramos un fallo y no somos capaces de entrar a nuestro correo.

Figura 6.12: Parámetros de conexión con Thunderbird

Tras realizar varias comprobaciones, nos vamos al fichero log del servidor, el cual se encuentra en /var/log/mail.log y nos encontramos con un fallo de certificados.


```
Jan 17 23:43:44 UbuntuServer dovecot: imap-login: Error: SSL: Stacked error: error:14094418:SSL routines:ssl3_read_byte:
es:tlsv1 alert unknown ca: SSL alert number 48
Jan 17 23:43:44 UbuntuServer dovecot: imap-login: Disconnected (disconnected before auth was ready, waited 0 secs): us
er=<>, rip=217.216.101.206, lip=89.36.222.93, TLS: SSL_read() failed: Unknown error, session=<68dUCFJGF0vZ2GX0>
Jan 17 23:43:44 UbuntuServer postfix/submission/smtpd[2074]: disconnect from 217.216.101.206.dyn.user.ono.com[217.216.
101.206] commands=0/0
```

Figura 6.13: Fallo encontrado en el log del servidor de correo.

Tras mucho *googlear* sin resultado, consulté a un administrador de sistemas qué podía estar fallando. Haciendo recuento de todo lo que hice para la instalación, me preguntó cómo había conseguido el certificado necesario para el uso de TLS, el cual, según el tutorial de DigitalOcean, se encuentra ya instalado en el VPS. Ahí se encontraba el error, puesto que es necesario obtener un certificado firmado por una autoridad de confianza. Buscando certificados gratuitos, encontré Let'sEncrypt [17]. Para obtener el certificado es necesario instalar el paquete *python-letsencrypt-apache* mediante apt y ejecutar el comando *letsencrypt certonly -d guilleando.tk -d www.guilleando.tk -d correo.guilleando.tk*, donde se usan los parámetros *certonly* y *-d* para indicar que solo se requiere el certificado (y no la instalación de este, se hará de manera manual) y los dominios a los que afectará, respectivamente. Dado que existen un gran número de certificados establecidos tanto en la configuración de postfix como de dovecot, no tuve más remedio que consultar una referencia de dudosa validez para establecerlos correctamente [18]. En ella, se explica que se deben de modificar los siguientes parámetros de */etc/postfix/main.cf* con la ubicación del certificado creado con Let'sEncrypt.

```
smtpd_tls_cert_file=/etc/letsencrypt/live/guilleando.tk/fullchain.pem
smtpd_tls_key_file=/etc/letsencrypt/live/guilleando.tk/privkey.pem
smtpd_use_tls=yes
```

Figura 6.14: Estableciendo los certificados en postfix.

Tendremos que realizar lo mismo en */etc/dovecot/conf.d/10-ssl.conf* para establecer correctamente los certificados en dovecot.

```
# certificate, just make sure to update the domains in dovecot.conf
ssl_cert = </etc/letsencrypt/live/guilleando.tk/fullchain.pem
ssl_key = </etc/letsencrypt/live/guilleando.tk/privkey.pem
```

Figura 6.15: Estableciendo los certificados en dovecot.

Tras ello, se reinician los servicios correspondientes a postfix y dovecot y se vuelve a probar Thunderbird, con la sorpresa de que ya podemos acceder e, incluso, enviar y recibir correos con normalidad.

Your name: Your name, as shown to others

Email address:

Password:

☒ Remember password

The following settings were found by probing the given server

	Server hostname	Port	SSL	Authentication
Incoming:	IMAP <input type="text" value="correo.guilleando.tk"/>	993 <input type="text"/>	SSL/TLS <input type="text"/>	Normal password <input type="text"/>
Outgoing:	SMTP <input type="text" value="correo.guilleando.tk"/>	587 <input type="text"/>	STARTTLS <input type="text"/>	Normal password <input type="text"/>

Username: Incoming: Outgoing:

Figura 6.16: Conexión válida de Thunderbird con el servidor de correo.

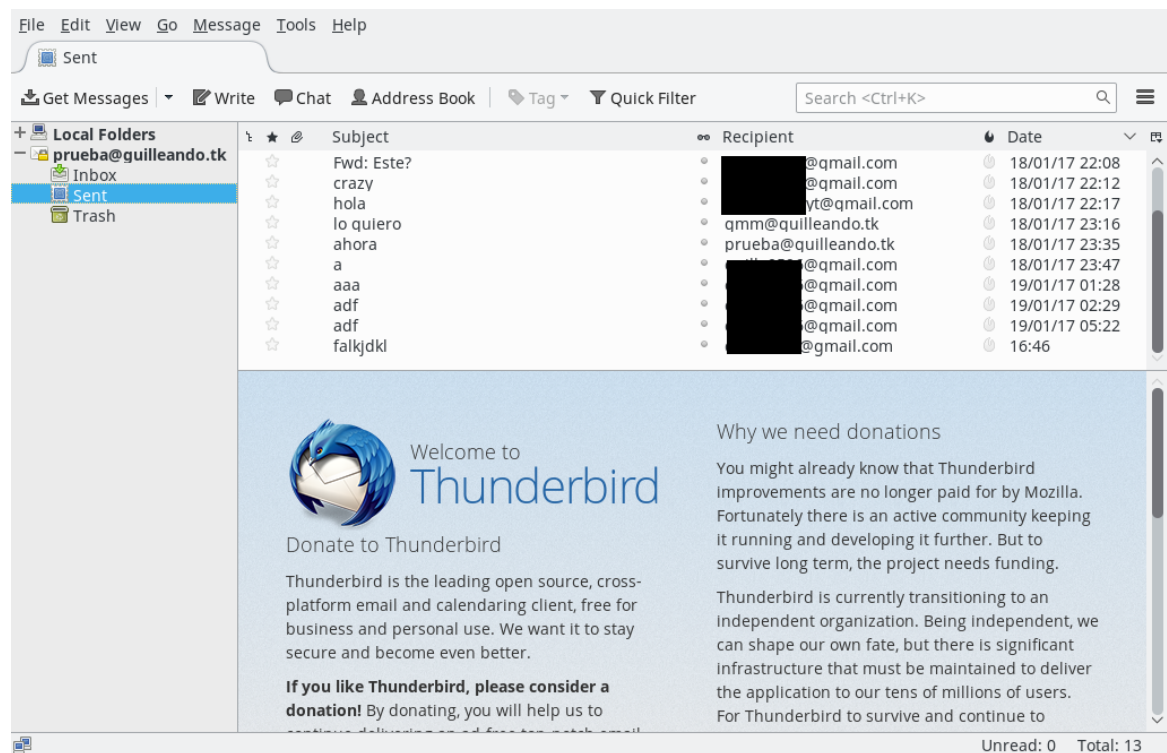


Figura 6.17: Correos enviados con la cuenta de prueba.



Figura 6.18: Correo recibido en cuenta de gmail.

Se ha intentado también la instalación de SquirrelMail [19], una aplicación basada en PHP que nos ofrece una interfaz web para acceder de manera sencilla a una cuenta de nuestro servidor de correo, pero su resultado no ha sido satisfactorio.

Referencias

- [1] RedHat. *Setting persistent tuning parameters*. Consultado el 3 de enero de 2017.
- [2] CentOS. *sysctl - Linux man page*. Consultado el 3 de enero de 2017.
- [3] CentOS. *Turning on Packet Forwarding*. Consultado el 3 de enero de 2017.
- [4] *Kernel parameters*. Consultado el 3 de enero de 2017.
- [5] Microsoft. *Cómo realizar una copia de seguridad y restaurar el registro de Windows*. Consultado el 3 de enero de 2017.
- [6] Moodle. *Performance recommendations*. Consultado el 3 de enero de 2017.
- [7] Moodle. *Apache performance - Performance recommendations*. Consultado el 3 de enero de 2017.
- [8] Moodle. *IIS performance - Performance recommendations*. Consultado el 3 de enero de 2017.
- [9] Microsoft. *Registry Value Types*. Consultado el 3 de enero de 2017.
- [10] Microsoft. *Optimizing IIS Performance*. Consultado el 3 de enero de 2017.
- [11] *curl - Linux man page*. Consultado el 3 de enero de 2017.
- [12] *ArubaCloud*. Consultado el 17 de enero de 2017.
- [13] *How To Configure a Mail Server Using Postfix, Dovecot, MySQL, and SpamAssassin*. Consultado el 17 de enero de 2017.
- [14] *The Postfix Home Page*. Consultado el 17 de enero de 2017.
- [15] *This POODLE bites: exploiting the SSL 3.0 fallback*. Consultado el 17 de enero de 2017.
- [16] *Dovecot - Secure IMAP Server*. Consultado el 17 de enero de 2017.

- [17] [*Let's Encrypt*](#). Consultado el 18 de enero de 2017.
- [18] [*Postfix and Dovecot on Ubuntu with a Let'sEncrypt SSL certificate*](#). Referencia no válida. Consultado el 18 de enero de 2017.
- [19] [*SquirrelMail - Webmail for nuts*](#). Consultado el 19 de enero de 2017.