



Proyecto IV

Paula Dolado | Gino Espejo | Iago Cabello

ICA0PRJ4I01 Administración de sistemas informáticos en la red.

Tabla de contenido

Implementación de la infraestructura	2
Organización direcciones ip	2
Descripción esquemática	2
DNS.....	3
Implementación y configuración del servicio de correo Zimbra	4
Ampliación 1: Envío de Correos	6
Ampliación 2: Active Directory	7
Ampliación 3: IMAP	9
Implementación y configuración del servicio de mensajería instantánea EJabberd.	10
Ampliación 1: Active Directory	14

Implementación de la infraestructura

Organización direcciones ip

Dispositivo	Dirección ip
Router	172.17.33.1/16
ESXI	172.17.33.2/16
Servido Zimbra	172.17.33.3/16
Servidor Jabber	172.17.33.4/16
Servidor Kodi	172.17.33.5/16
Servidor Elastix	172.17.33.6/16
Active Directory	172.17.33.7/16
DNS	172.17.33.8/16
Cliente Windows	172.17.33.9/16
Zabbix	172.17.33.10/16
Switch 1	172.17.33.11/16
Switch 2	172.17.33.12/16

Descripción esquemática

Para nuestra infraestructura hemos decidido colgar todas las maquinas desde la red troncal, por lo tanto, todas las maquinas tendrán conexión a las otras sin ninguna regla que las segmente. Hemos partido de 3 máquinas donde:

- 1 será el servidor de correo electrónico.
- 2 será un entorno virtualizado de ESXI con el Active directory, Kodi, Dns, Zabbix, enrutador i Elastix.
- 3 será el Servidor de Jabber.
- Como clientes windows tendremos nuestros propios equipos de trabajo.

Nuestro único enrutador hará la función de permitir el paso para poder conectarnos a través de un Túnel SSH.

Para terminar con las conexiones remotas utilizaremos un servidor externo y utilizaremos una máquina virtual para hacer de Cliente al igual que nuestra maquina local.

DNS

Como servidor DNS crearemos una máquina virtual dentro de un ESXI de vsphere, este será instalado en un Ubuntu 22.04 a través del webmin.

Una vez tenemos el plugin de Servidor DNS crearemos una Zona maestra con el nombre de nuestro dominio. En la imagen a continuación podremos ver nuestro registro de máquinas junto a su dirección ip, como veremos más adelante ns1 hace referencia a la zona en cuestión mientras que ns2 y ns3 son las zonas esclavas.

Es importante destacar la línea 14 ya que trata de un registro mx, necesario para el funcionamiento de nuestro servidor de correo Zimbra puesto que enlaza esté a nuestro dominio.

```
1 $ttl 3600
2 pig.loc. IN SOA ns1.pig.loc. root.pig.loc. (
3     2023051800
4     3600
5     600
6     1209600
7     3600 )
8 pig.loc. IN NS ns1.pig.loc.
9 pig.loc. IN NS ns2.pig.loc.
10 pig.loc. IN NS ns3.pig.loc.
11 pig.loc. IN NS zimbra-g2.pig.loc.
12
13 ; MX Record
14 pig.loc. IN MX 10 zimbra-g2.pig.loc.
15
16 ; Machine Names
17 pig.loc. IN A 172.17.33.8
18 ns1.pig.loc. IN A 172.17.33.8
19 ns2.pig.loc. IN A 172.17.9.254
20 ns3.pig.loc. IN A 172.17.9.255
21 zimbra-g2.pig.loc. IN A 172.17.33.3
22 ejabberd.pig.loc. IN A 172.17.33.4
23 ldap.pig.local. IN A 172.17.33.7
```

Continuando con la configuración del servidor deberemos añadir las direcciones de nuestras dos zonas esclavas como forwards, esto reenviará las solicitudes a los otros DNS si no conoce la respuesta para que gestionen la consulta.

☆ Forwarding and Transfers

Global forwarding and zone transfer options

Servers to forward queries to

IP address	Port
172.17.9.254	<input checked="" type="radio"/> Default <input type="radio"/>
172.17.9.255	<input checked="" type="radio"/> Default <input type="radio"/>

☆ Zone Options

pig.loc

Zone Options

Check names? ☐ Warn ☐ Fail ☐ Ignore ☒ Default

Allow updates from...
any

Allow queries from...
any

Notify slaves of changes? ☒ Yes ☐ No ☐ explicit ☐ Default

Allow transfers from...
172.17.9.254
172.17.9.255

Also notify slaves...
172.17.9.254
172.17.9.255

Por otro lado, al crear las zonas esclavas deberemos introducir la dirección ip de la zona maestra debido a que a raíz de una transferencia de archivos recogerá automáticamente todos los registros que ese contiene.

Servidores Maestros	172.17.33.8
---------------------	-------------

Como veremos en la imagen siguiente nuestra zona esclava coge el registro del servidor de correo de manera automática.

Servidor de Correo Registros			
En pig.loc			
Mostrar registros coincidentes:		<input type="text" value="Buscar"/>	
Nombre	TTL	Prioridad	Servidor de Correo
pig.loc.	3600	10	zimbra-g2.pig.loc.

Implementación y configuración del servicio de correo Zimbra

Zimbra es un programa colaborativo con el servicio principal de un servidor de correo electrónico además de tener calendarios, documentos y contactos entre otras cosas, para ello haremos uso de una imagen de un Ubuntu 20.04.¹

Previamente a la instalación deberemos de tener el archivo de /etc/hosts con la siguiente estructura:

```
pig@zimbra-g2:~$ cat /etc/hosts
127.0.0.1 localhost
127.0.1.1 zimbra-g2
172.17.33.3 zimbra-g2.pig.local
```

Como primer paso para para obtener el servidor de correo instalaremos el paquete de aplicaciones de LibreOffice y descargaremos la penúltima versión del Zimbra. (NOTA: La versión más actual 9, no tiene la versión de trial para poder hacer una pequeña demostración de manera gratuita ya que para implementarlo se deberá tener obligatoriamente una licencia.)

```
# apt-get install libreoffice libreoffice-l10n-* fonts-vlgothic

# wget https://files.zimbra.com/downloads/9.0.0_BETA/zcs-NETWORK-9.0.0_BETA_4153.UBUNTU20_64.20210920031140.tgz
```

¹ <https://computingforgeeks.com/how-to-install-zimbra-collaboration-on-ubuntu/>
<https://help.clouding.io/hc/es/articles/360011382980-C%C3%B3mo-instalar-Zimbra-9-en-Ubuntu-20-04-LTS>

Descomprimos el paquete descargado y nos adentramos para ejecutar el script de instalación

```
# tar -xzf zcs*

#cd zcs-NETWORK-9.0.0_BETA_4153.UBUNTU20_64.20210920031140/

# ./install.sh
```

A raíz de aquí se iniciará el asistente para instalar el Zimbra, tendremos que aceptar los términos y condiciones además de aceptar utilizar el repositorio de Zimbra. Veremos por pantalla una lista de paquetes que instalar y dejará seleccionar los que queramos tener, para entonces desmarcaremos el DNS caché.

Una vez finalizada la instalación deberemos indicarle una contraseña para el usuario de administración, ese mismo usuario será con el que accedamos vía web con los puertos mostrados en este mismo menú.

```
Store configuration

1) Status: Enabled
2) Create Admin User: yes
3) Admin user to create: admin@pig.loc
4) Admin Password: set
5) Anti-virus quarantine user: virus-quarantine.kvidgsprm@pig.loc
6) Enable automated spam training: yes
7) Spam training user: spam.5fr4_f9ils@pig.loc
8) Non-spam(Ham) training user: ham.qxrc7xhr0r@pig.loc
9) SMTP host: zimbra-g2.pig.local
10) Web server HTTP port: 8080
11) Web server HTTPS port: 8443
12) Web server mode: https
13) IMAP server port: 7143
14) IMAP server SSL port: 7993
15) POP server port: 7110
16) POP server SSL port: 7995
```

Para cerrar la configuración saldremos con la tecla r al menú inicial donde se podrá verificar que tenemos todos los módulos habilitados, una vez conformes guardaremos la configuración para que surja efecto.

```
Main menu

1) Common Configuration:
2) zimbra-ldap: Enabled
3) zimbra-logger: Enabled
4) zimbra-mta: Enabled
5) zimbra-snmp: Enabled
6) zimbra-store: Enabled
7) zimbra-spell: Enabled
8) zimbra-proxy: Enabled
9) zimbra-imapd: Enabled
10) Default Class of Service Configuration:
    s) Save config to file
    x) Expand menu
    q) Quit

*** CONFIGURATION COMPLETE - press 'a' to apply
Select from menu, or press 'a' to apply config (? - help) a
Save configuration data to a file? [Yes] Yes
Save config in file: [/opt/zimbra/config.137085]
Saving config in /opt/zimbra/config.137085...done.
The system will be modified - continue? [No] Yes
```



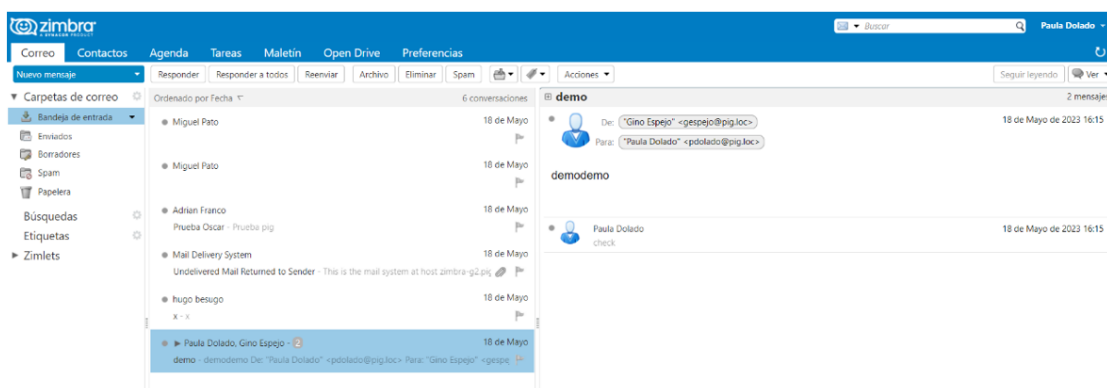
Ampliación: Envío de Correos

Empezando en la consola de administración crearemos 3 usuarios de prueba correspondiendo a los miembros del grupo, para ello deberemos indicar el nombre y apellido que se muestra, usuario y contraseña.

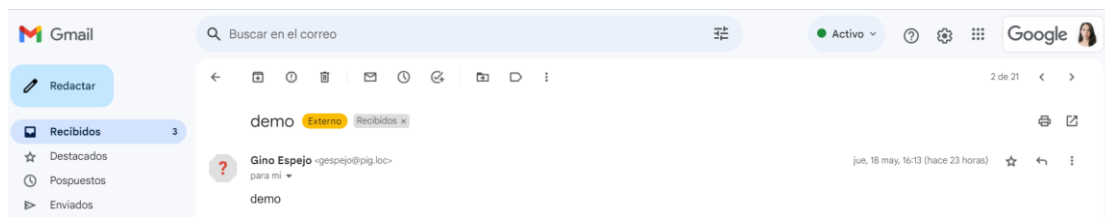


Si iniciamos sesión con uno de los usuarios haremos las siguientes pruebas:

- Envío y recepción de correo dentro de la organización interna.



- Envío y recepción de correo dentro de la organización externa.

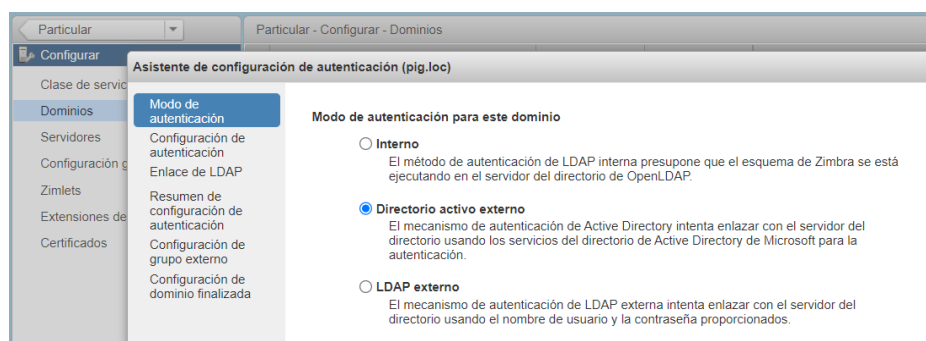


En este caso hemos enviado al correo de la escuela, pero en la captura de pantalla anterior también podremos ver hacia los dominios de nuestros compañeros.

Ampliación 2: Active Directory

Continuando con la segunda ampliación implementaremos los usuarios del Active Directory en el mismo Zimbra.

Para conseguirlo, nos desplazaremos a la pantalla de administración de nuevo y seleccionaremos nuestro dominio y con un clic derecho abriremos el asistente de configuración de autenticación para pig.loc, seleccionaremos AD i le introduciremos nuestros datos como nombre del dominio, dirección ip del controlador de dominio y un usuario con permisos de administrador.



Resumen de configuración autenticación

Mecanismo de autenticación: **Directorio activo externo**

Nombre del dominio AD: pig.local

URL de LDAP: ldap://172.17.33.7:3268

Introduce un nombre de usuario y una contraseña para probar la configuración de autenticación

Usuario: administrador

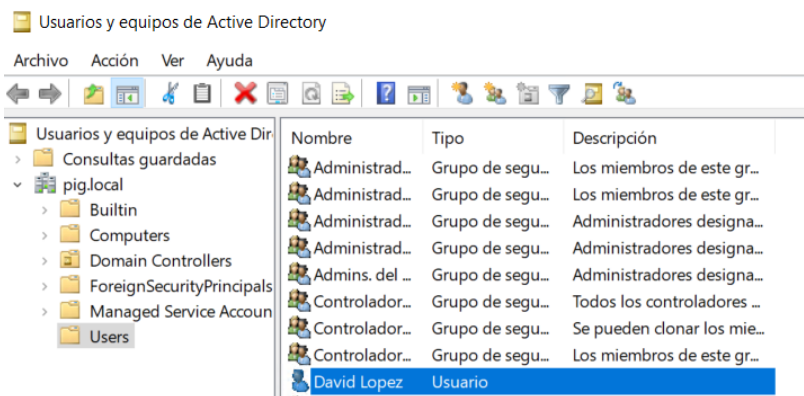
Contraseña:

Prueba

Prueba de autenticación satisfactoria

DN de enlace
computado: administrador@pig.local

Por otro lado, en la administración de usuarios y grupos de nuestro controlador de dominio crearemos un usuario con información básica.



Para añadir este usuario a nuestro Zimbra deberemos hacer la acción de crear un nuevo usuario con su nombre y apellido que mostrar y su nombre de usuario.

— Nombre de cuenta —

Nombre de cuenta: * d.lopez @ pig.loc

Nombre: David

Inicial 2º nombre:

Apellido: * Lopez

Nombre mostrado: David Lopez ☒ auto

☐ Ocultar en GAL:

A diferencia de los usuarios creados anteriormente para implementarlo con el active directory deberemos indicarle la cuenta mediante LDAP, es decir, la ruta donde se encuentra creado con la estructura que se muestra

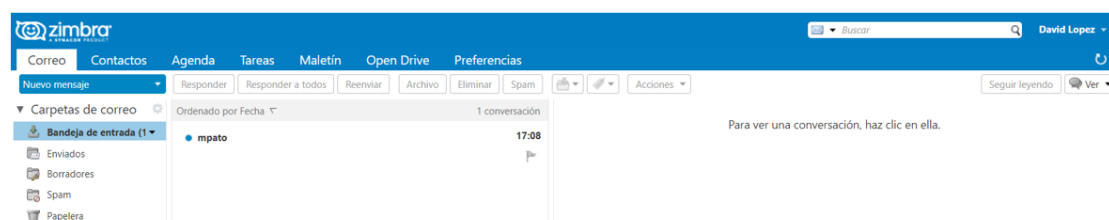
— Autenticación externa —

Cuenta LDAP externa para autenticación: CN=David Lopez,CN=Users,DC=pig,DC=local

— Configuración de zona horaria —

Zona horaria: GMT +01:00 Bruselas, Copenhague, Madrid, París

A partir de aquí, los usuarios locales que creamos a no ser que estén en el AD no se podrán utilizar además de los cambios en la configuración de los usuarios existentes en el AD se verán afectados en la configuración del Zimbra como por ejemplo un cambio de contraseña desde el controlador de dominio.



Ampliación 3: IMAP

IMAP permite acceder al correo desde cualquier dispositivo u otra aplicación vale la pena comentar que este en realidad no se descarga ni almacena de manera local su función trata de leer el servidor de correo convirtiéndose así en un protocolo no crítico para su funcionamiento. Por otro lado, el protocolo SMTP utiliza tanto nuestro servidor Zimbra como otros para enviar y recibir e-mails.

Paralelamente al IMAP tenemos el protocolo de POP3 este almacena temporalmente hasta que son descargados por el usuario. Por lo anterior, podemos concluir que la diferencia entre ambos es el almacenamiento.

Continuando con la tercera ampliación vamos a implementar una cuenta Zimbra en la aplicación de mensajera de Microsoft, Outlook.

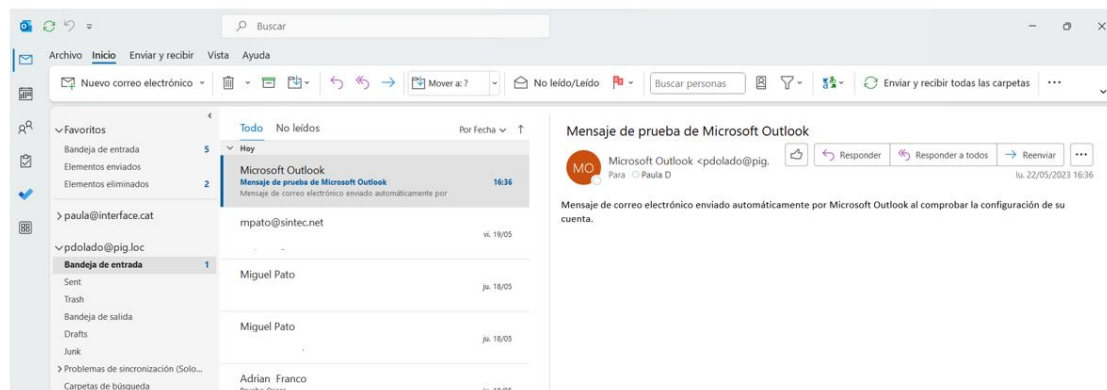
Para ello no se deberá modificar nada de la configuración del propio Zimbra puesto a que por defecto todas las cuentas tienen habilitada la extensión de conexión mediante IMAP o POP externo.

Si hay que tener en cuenta la configuración desde el perfil de mail a través del panel de control puesto a que seleccionando la opción de imap deberemos indicar los puertos siguientes para poder permitir la conexión con el servidor.

Al agregar la cuenta es necesario indicar tanto como servidor entrante como saliente el nombre de máquina de nuestro servidor Zimbra. Además, utilizando la conexión cifrada de IMAP y SMTP será mediante los puertos 7993 y 465 respectivamente.

Las imágenes muestran la configuración de una cuenta de correo en Outlook. La primera imagen es la ventana 'Agregar cuenta' con la pestaña 'Configuración de cuenta IMAP y POP' seleccionada. Se ingresan los datos de usuario: nombre 'Paula D', dirección de correo 'pdolado@pig.loc'. En la sección 'Información del servidor', se selecciona 'IMAP' como tipo de cuenta, y se ingresan 'zimbra-g2.pig.loc' tanto para el servidor entrante como para el servidor saliente (SMTP). En la sección 'Información de inicio de sesión', se ingresan el nombre de usuario 'pdolado@pig.loc' y una contraseña, con la opción 'Recordar contraseña' marcada. La segunda imagen muestra la ventana 'Configuración de correo electrónico de Internet' con la pestaña 'Avanzadas' seleccionada. Se marca la opción 'Mi servidor de salida (SMTP) requiere autenticación' y se selecciona 'Utilizar la misma configuración que mi servidor de correo de entrada'. En la sección 'Números de puerto del servidor', se ingresan los puertos 7993 para el servidor de entrada (IMAP) y 465 para el servidor de salida (SMTP). Ambas configuraciones de conexión cifrada están establecidas en 'SSL/TLS'.

Una vez se compruebe la conexión exitosa seguramente aparezca una ventana emergente ya que el certificado no es seguro, cuando la aceptemos ya podremos iniciar la aplicación Outlook y producir desde esta.



Implementación i configuración del servicio de mensajería instantánea EJabberd.

Es un servidor de mensajería instantánea Jabber/XMPP, licenciado bajo GPLv2 y escrito en Erlang. Entre otras características, es multiplataforma, tolerante a fallas, clusterizable y modular.

Entre los módulos más interesantes para entornos corporativos se encuentran el de autenticación contra LDAP y el vCARD LDAP, lo cual elimina la complicación de crear usuarios a mano.

Si bien su fácil integración con Jappix lo hacen una opción bastante interesante, Openfire es otra opción relativamente más sencilla de configurar y de mantener configurada.

En nuestro entorno hemos decidido instalar nuestro servidor de mensajería en un sistema operativo Linux.

Requisitos mínimos

- RAM: 2GB
- CPU: 2 Cores
- DISCO: 20 GB
- SO: Linux (Ubuntu 22.04)

Prerrequisitos

Es necesario cambiar el nombre de nuestro servidor por el nombre de dominio (más adelante se explicará la razón).

```
# hostnamectl set-hostname <nombre_dominio> --static  
# reboot
```

Instalación

Descargamos la última versión disponible para nuestro sistema operativo.

Podemos visitar la [página web oficial](#) y adaptar la instrucción para obtener la última versión.

```
# wget https://github.com/processone/ejabberd/releases/download/23.04/ejabberd_23.04-1_amd64.deb
```

Una vez tenemos descargado el archivo .deb pasamos a instalarlo y acto seguido detenemos el servicio.

```
# dpkg -i ejabberd_23.04-1_amd64.deb
# systemctl stop ejabberd
```

Configuración

Una vez hecho esto pasamos a editar el archivo /opt/ejabberd/conf/ejabberd.yml y añadimos/modificamos algunas líneas para configurar nuestro ejabberd.

Dar permisos de administrador

```
acl:
  admin:
    user: admin1@example.org
```

```
access_rules:
  configure:
    allow: admin
```

- Hecho esto iniciamos el servicio y ya podemos iniciar sesión a través de la dirección IP

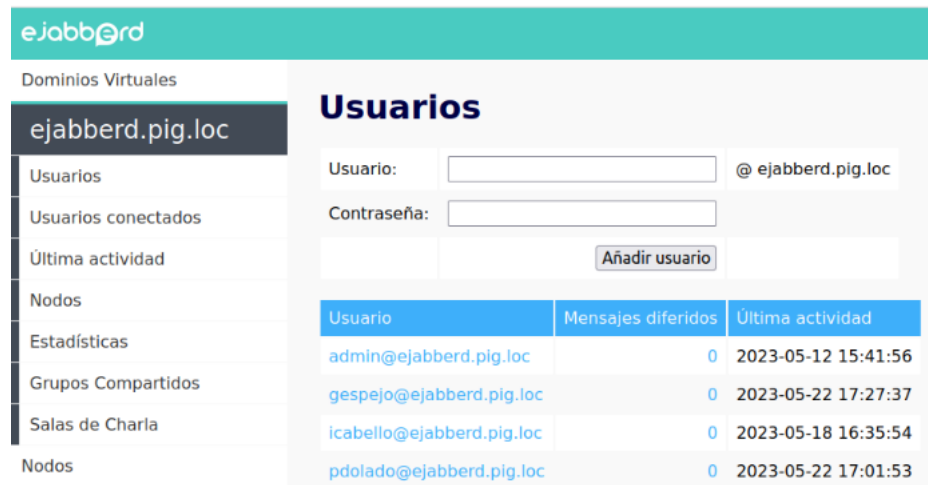
```
# systemctl start ejabberd
```

```
● ejabberd.service - XMPP Server
   Loaded: loaded (/etc/systemd/system/ejabberd.service; enabled; vendor preset: enabled)
   Active: active (running) since Mon 2023-05-22 15:15:48 UTC; 23h ago
```



Registrar usuarios

- Vía comando: # ejabberdctl register <user> <dominio> <password>
- Por web



The screenshot shows the ejabberd web interface. On the left is a sidebar with a teal header 'ejabberd' and a list of menu items: 'Dominios Virtuales', 'ejabberd.pig.loc', 'Usuarios', 'Usuarios conectados', 'Última actividad', 'Nodos', 'Estadísticas', 'Grupos Compartidos', 'Salas de Charla', and 'Nodos'. The main area is titled 'Usuarios' and contains a registration form with fields for 'Usuario:' and 'Contraseña:', a dropdown for '@ ejabberd.pig.loc', and an 'Añadir usuario' button. Below the form is a table with three columns: 'Usuario', 'Mensajes diferidos', and 'Última actividad'.

Usuario	Mensajes diferidos	Última actividad
admin@ejabberd.pig.loc	0	2023-05-12 15:41:56
gespejo@ejabberd.pig.loc	0	2023-05-22 17:27:37
icabello@ejabberd.pig.loc	0	2023-05-18 16:35:54
pdolado@ejabberd.pig.loc	0	2023-05-22 17:01:53

Existen diversos clientes que aceptan servidores XMPP, sin embargo, en nuestro entorno de prácticas hemos probado solo 2.

Entre los más conocidos tenemos:

- Pidgin
- Psi +
- Jitsi

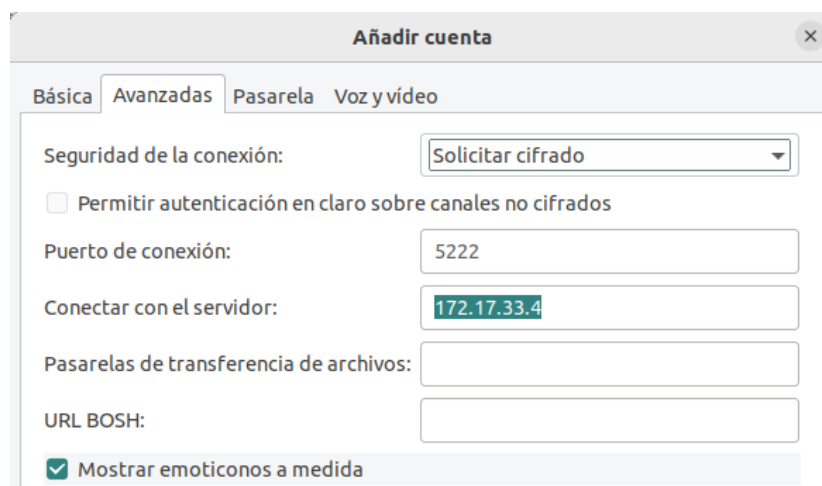
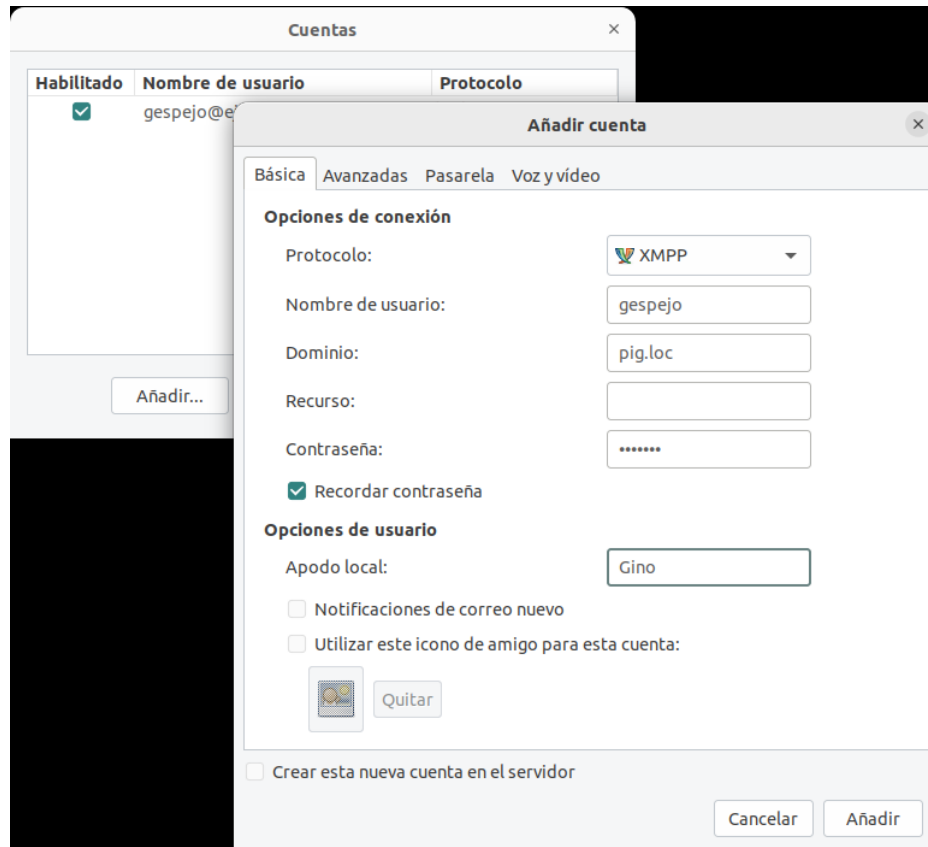
Para nuestro ejemplo vamos a configurar solamente una aplicación, todos básicamente se configuran de la misma manera.

Configuración Pidgin

Podemos descargarnos la aplicación desde su [web oficial](#).

La instalación depende de nuestro sistema operativo, en Windows te descargas el .exe e instalas sin mayores problemas y en Linux se puede instalar mediante el comando **apt install pidgin**.

Una vez instalado nos aparece un mensaje para poder añadir nuestro primero usuario. Adjuntaré algunas capturas ya que es más fácil de entender la configuración.



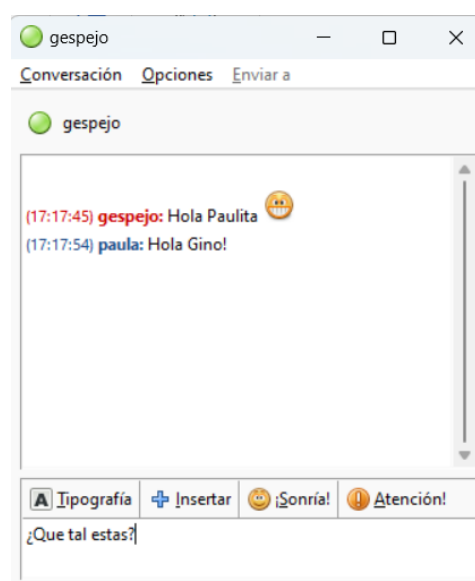
Añadir amigos

Ahora debemos añadir “buddies o amigos”, para poder iniciar la conversación con alguien de nuestro server.

Podemos hacerlo desde el menú principal Amigos -> Añadir un amigo, o con el acceso directo “CTRL + B”

Nos aparecerá la siguiente ventana y podemos añadir rellenando las siguientes opciones.

Añadido a nuestro amigo podemos iniciar una conversación.



Ampliación 1: Active Directory

Una de las ventajas de nuestro servidor Ejabberd es que podemos configurar la autenticación hacia un controlador de dominio a través del protocolo LDAP.

Para hacerlo debemos añadir las siguientes líneas en nuestro archivo ejabberd.yml

```
## Authentication method
auth_method: ldap
## DNS name of our LDAP server
ldap_servers:
  - 172.17.33.7
## Bind to LDAP server as "cn=Manager,dc=example,dc=org" with password
"secret"
ldap_rootdn: "CN=Administrador,CN=Users,DC=<domain>,DC=loc"
ldap_password: "*****"
ldap_encrypt: none
ldap_port: 389
## Define the user's base
```

```
ldap_base: "DC=<domain>,DC=loc"  
ldap_uids: {"sAMAccountName": "%u"}
```

Una vez añadido estas líneas necesitamos reiniciar el servicio
systemctl restart ejabberd