/

**Virtualización de servicios en Docker**

**Coordinador**

**Antonio Sepúlveda Ruiz**

**Grupo de trabajo**

**Luis Valderas Torres**

**Daniel De Tomas Plaza**

**Daniel Palencia Origüel**

Contenido

[Proyecto Docker 4](#_Toc38528899)

[Instalación De Docker 5](#_Toc38528900)

[Instalación De Openldap en docker 8](#_Toc38528901)

[Instalación de Interfaz Web Openldap en docker 9](#_Toc38528902)

[Creación y configuración de Usuario, Grupos y Unidades Organizativas en LDAP en docker 10](#_Toc38528903)

[Para justificar el usar Docker, hemos replicado el mismo proceso en el propio sistema Debian, sin usar Docker, para ver como Docker nos ayuda hacer despliegues mucho más rápido 15](#_Toc38528904)

[Instalación OpenLdap Debian puro 15](#_Toc38528905)

[Creación y configuración de Usuario, Grupos y Unidades Organizativas en LDAP en Debian puro 19](#_Toc38528906)

[Unidades organizativas 21](#_Toc38528907)

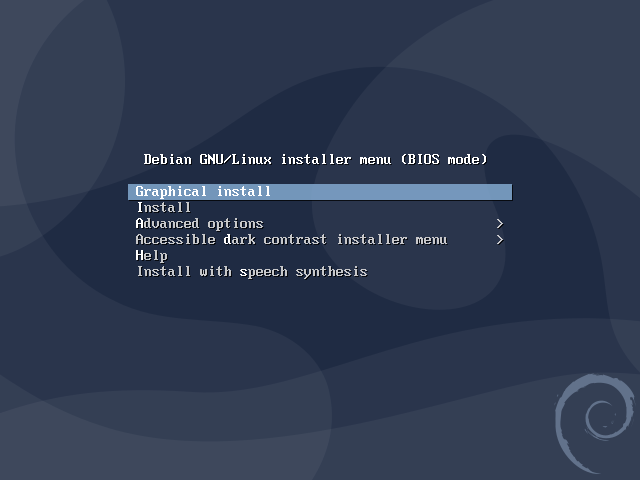
[Grupos 21](#_Toc38528908)

[Usuarios 22](#_Toc38528909)

[Instalación De Interfaz Web Openldap en Debian puro 24](#_Toc38528910)

# Proyecto Docker

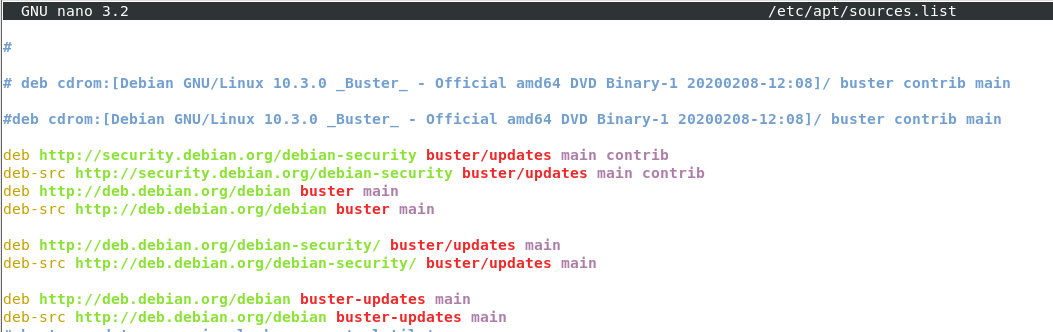
Para empezar el proyecto, lo primero que tenemos que hacer es simular la maquina de la empresa que va a contener Docker y sus servicios, para ello vamos a descargar una ISO de Debian en concreto la versión Debian 10.3.0 de su página web ([www.debian.org](http://www.debian.org)).



Una vez descargada usando VMware vamos a crear una máquina virtual con Debian que simulara ser el equipo de la empresa, vamos a configurar el nombre del equipo “Debian-Docker” y como usuario “Proyecto”.

Una vez instalado procedemos a añadir los repositorios de debian para así poder instalar/actualizar paquetes utilizando el siguiente comando.

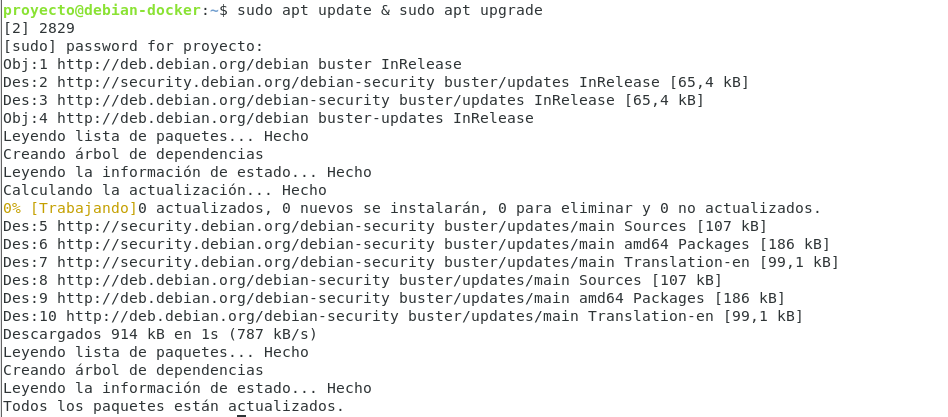
sudo nano /etc/apt/sources.list



En la web de debian podremos obtener la lista con los repositorios correspondientes.

Una vez añadidos los repositorios actualizaremos la lista de paquetes disponibles y actualizaremos los paquetes que no estén actualizados utilizaremos los siguientes comandos.

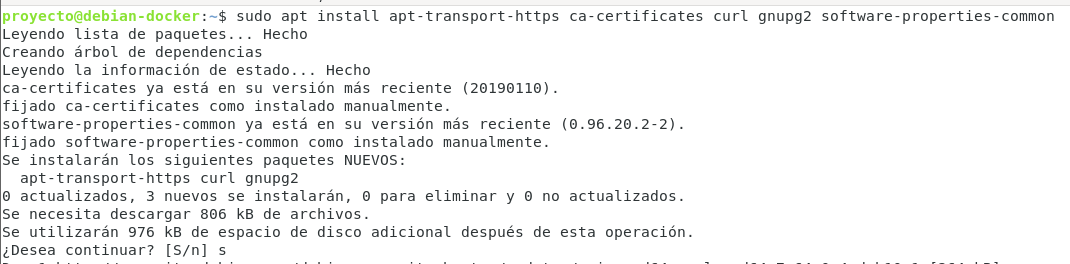
sudo apt update & sudo apt upgrade



## Instalación De Docker

Con el siguiente comando vamos a permitir a apt usar paquetes a través de HTTPS.

sudo apt install apt-transport-https ca-certificates curl gnupg2 software properties-common



Procedemos a añadir la clave GPG del repositorio oficial de Docker para poder descargar el paquete.

curl -fsSL https://download.docker.com/linux/debian/gpg | sudo apt-key add –



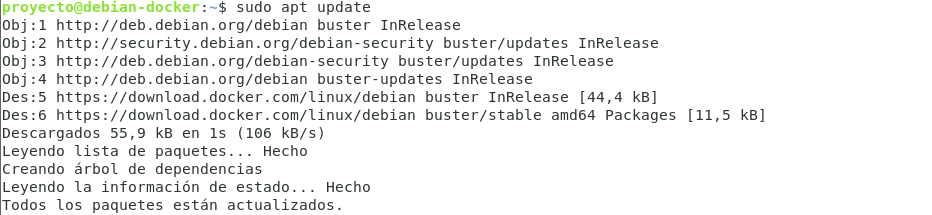
Ahora añadimos el repositorio de Docker alas fuentes del APT.

sudo add-apt-repository "deb [arch=amd64] https://download.docker.com/linux/debian $(lsb\_release -cs) stable"



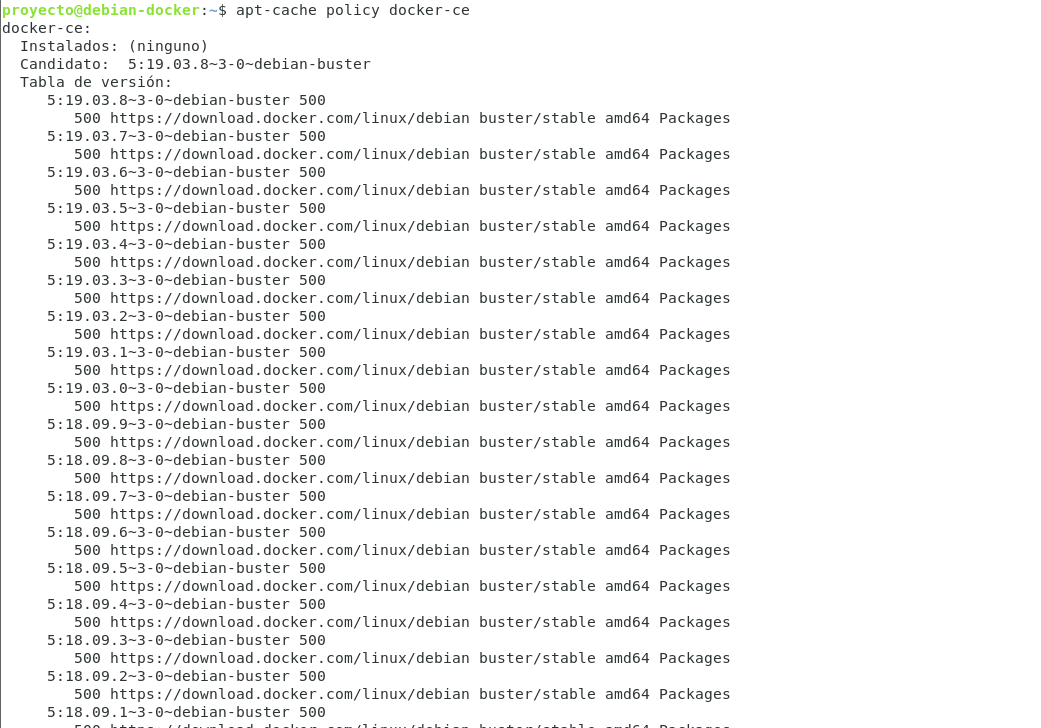
Volvemos a actualizar la base de datos de paquetes para que se añadan los datos de Docker.

sudo apt update



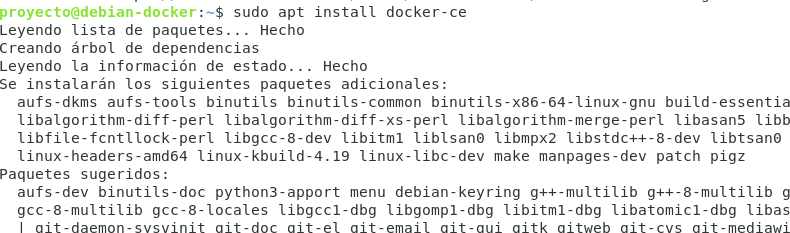
Verificamos que la instalación de Docker se va a realizar desde el repositorio de Docker en lugar del repositorio predeterminado de Debian.

apt-cache policy docker-ce



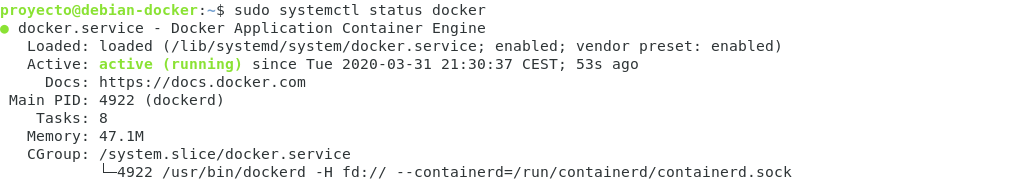
Procedemos a instalar Docker.

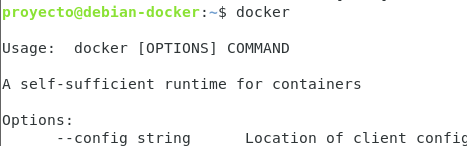
sudo apt install docker-ce



Verificamos que Docker se ha instalado correctamente.

sudo systemctl status Docker

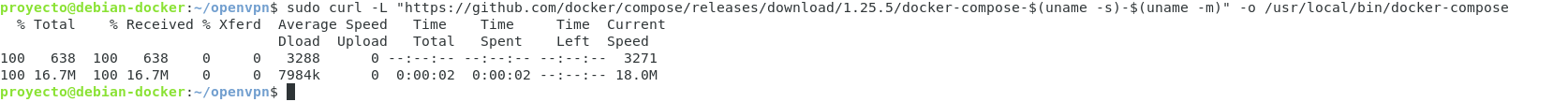




### Instalación Docker-Compose

Vamos a instalar Docker-Compose, que es una herramienta que permite realizar scripts para facilitar la creación de containers y servicios.Para instalar Docker-Compose procedemos a usar el siguiente comando.

sudo curl -L "https://github.com/docker/compose/releases/download/1.25.5/docker-compose-$(uname -s)-$(uname -m)" -o /usr/local/bin/docker-compose



Ahora le tenemos que dar permisos de ejecución para ello ejecutaremos el siguiente comando.

sudo chmod +x /usr/local/bin/docker-compose



Por último, para comprobar que Docker-compose se ha instalado sin problemas lo verificaremos con el siguiente comando.

Docker-compose --version

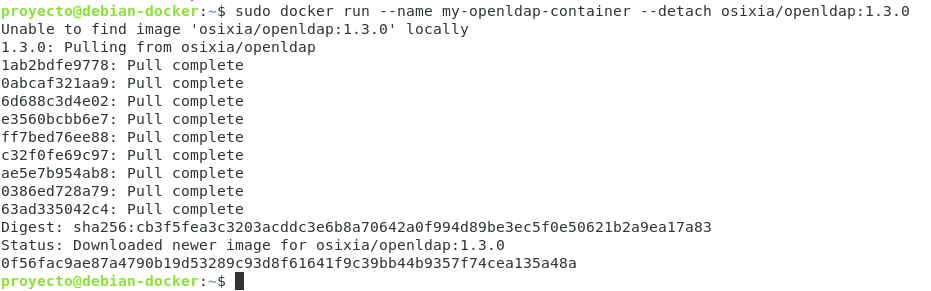


## Instalación De Openldap en docker

Para empezar, descargaremos la imagen de Docker Openldap

Docker run –name my-openldap-container –detach osixia/openldap:1.3.0

Obviamente como no lo tenemos en local, se conectará a buscar la imagen



A continuación, vamos a añadir nuestro dominio y la configuración principal con este comando

docker run -p 389:389 --name ldap-service --hostname ldap-service --env LDAP\_ORGANISATION="Jovellanos" --env LDAP\_DOMAIN="proyecto.docker" \

--env LDAP\_ADMIN\_PASSWORD="1234" --env LDAP\_BASE\_DN="dc=proyecto,dc=docker" --volume /data/slapd/database:/var/lib/ldap \

--volume /data/slapd/config:/etc/ldap/slapd.d --detach osixia/openldap:1.3.0

## Instalación de Interfaz Web Openldap en docker

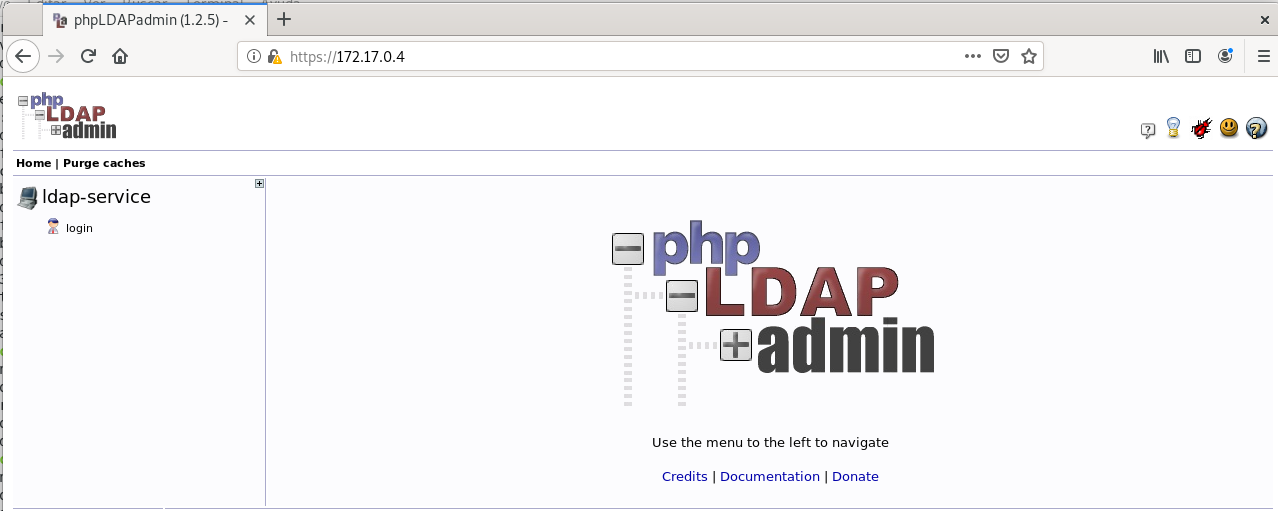
También para mayor comodidad vamos añadir una interfaz web para nuestro openldap

docker run --name phpldapadmin-service --hostname phpldapadmin-service --link ldap-service:ldap-host --env PHPLDAPADMIN\_LDAP\_HOSTS=ldap-service --detach osixia/phpldapadmin:0.9.0

Después con este comando averiguaremos cual es la ip asociada al proceso de ldapadmin

docker inspect -f "{{ .NetworkSettings.IPAddress }}" phpldapadmin-service

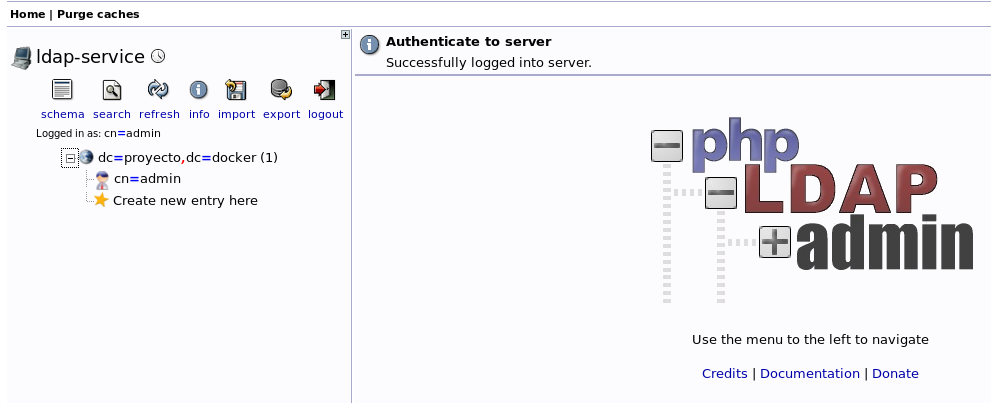
A continuación, si nos conectamos a esa ip accederemos a la interfaz web



Le daremos a la ventana de login poniendo los datos de nuestro dominio anteriormente creado

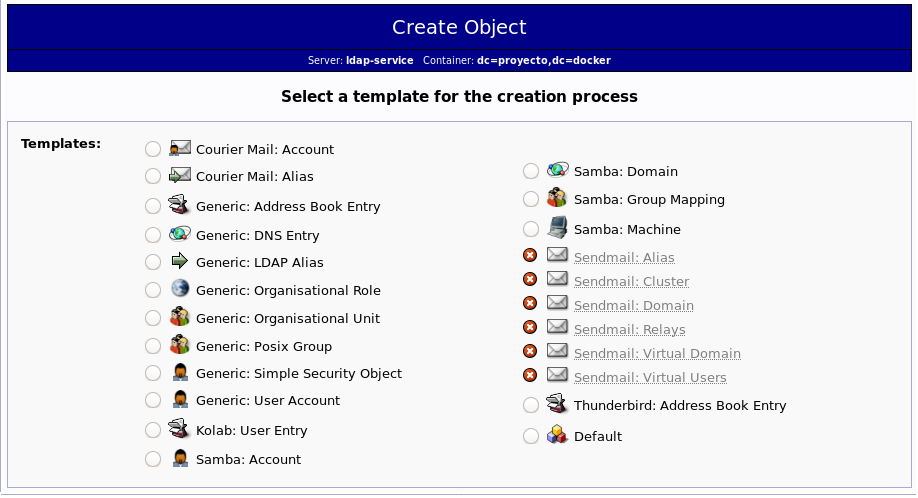


Y como comprobamos podemos acceder perfectamente

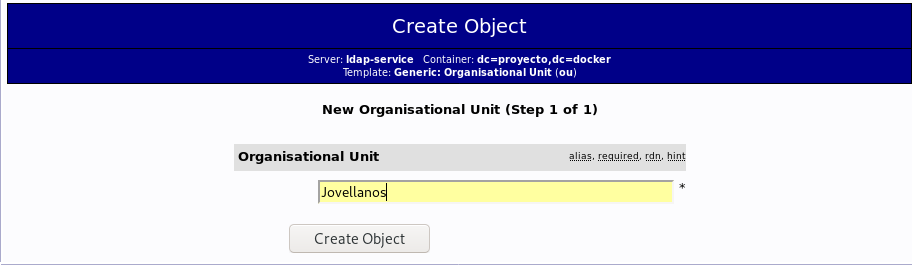


## Creación y configuración de Usuario, Grupos y Unidades Organizativas en LDAP en docker

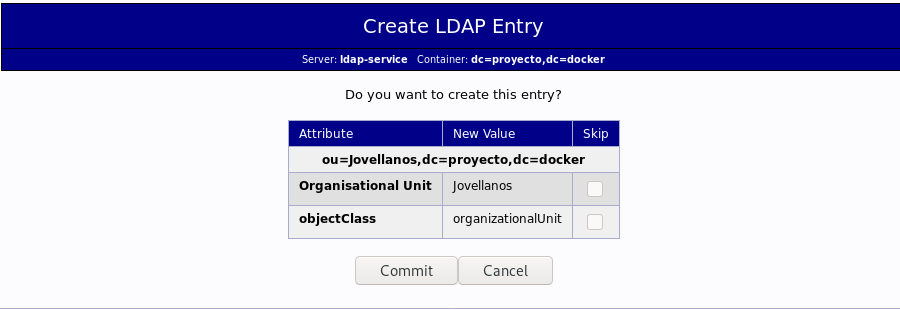
A continuación, creamos un par de entradas para comprobar su funcionalidad



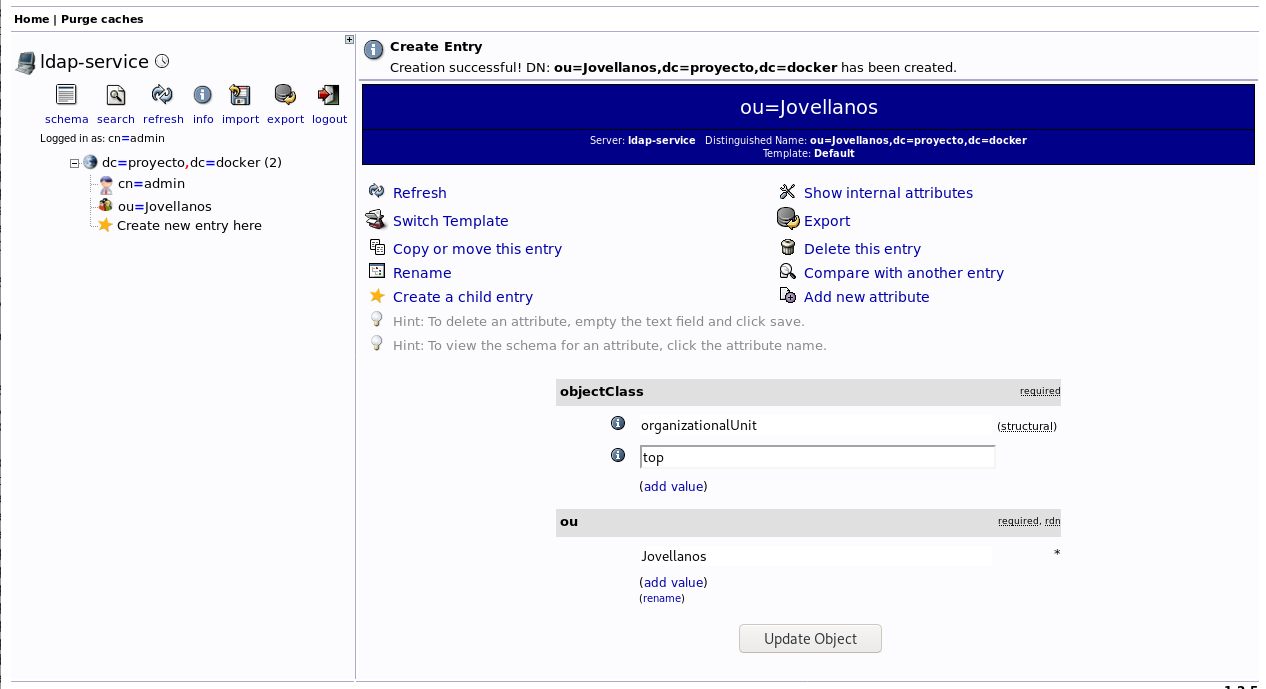
Para empezar, cogeremos crear una UO



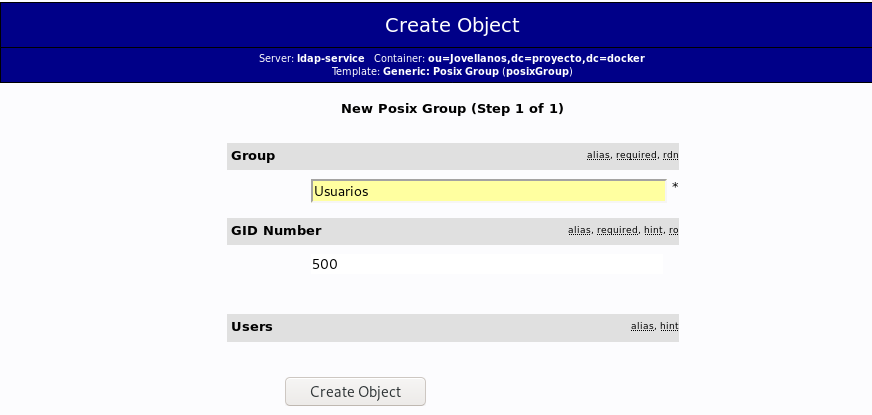
Comprobamos los datos metidos y le damos a commit



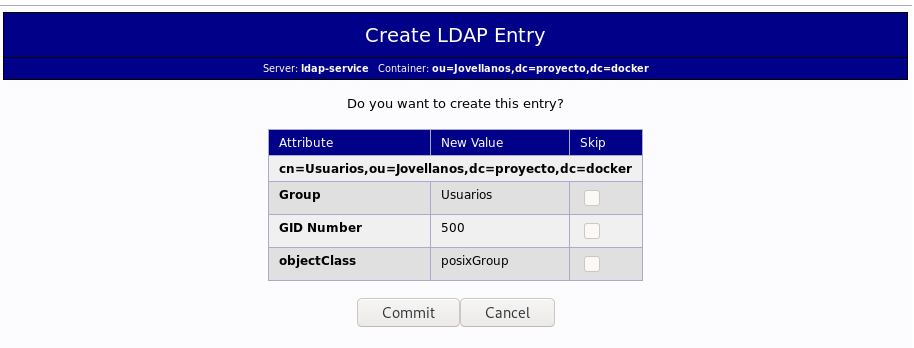
Como vemos, ya aparece a la izquierda



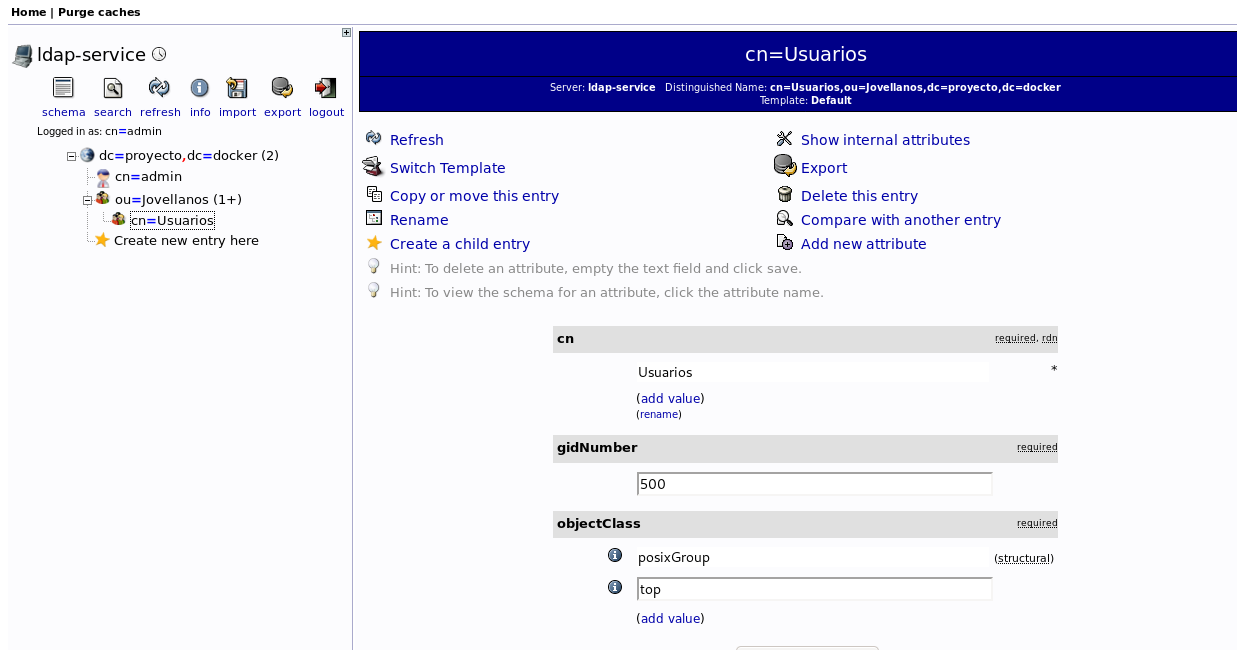
Creamos ahora un grupo llamado usuarios dentro de la UO Jovellanos



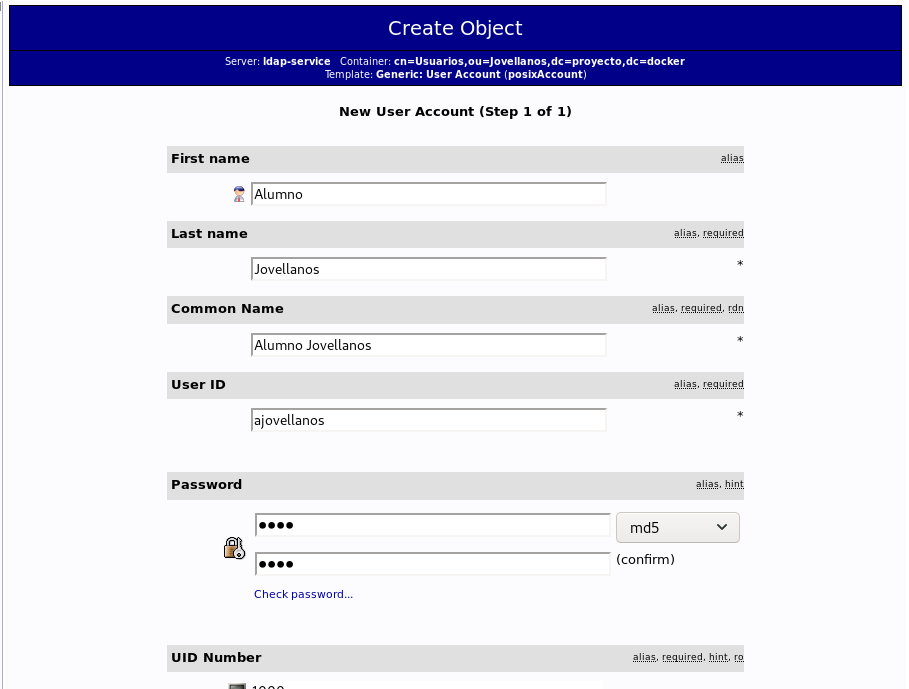
Comprobamos la entrada

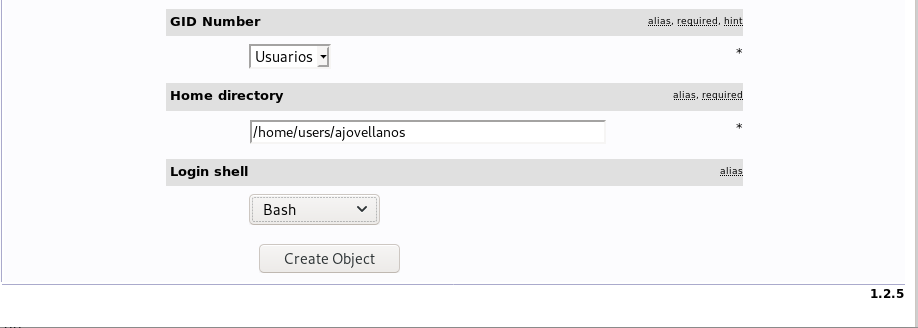


Y ya lo tenemos creado

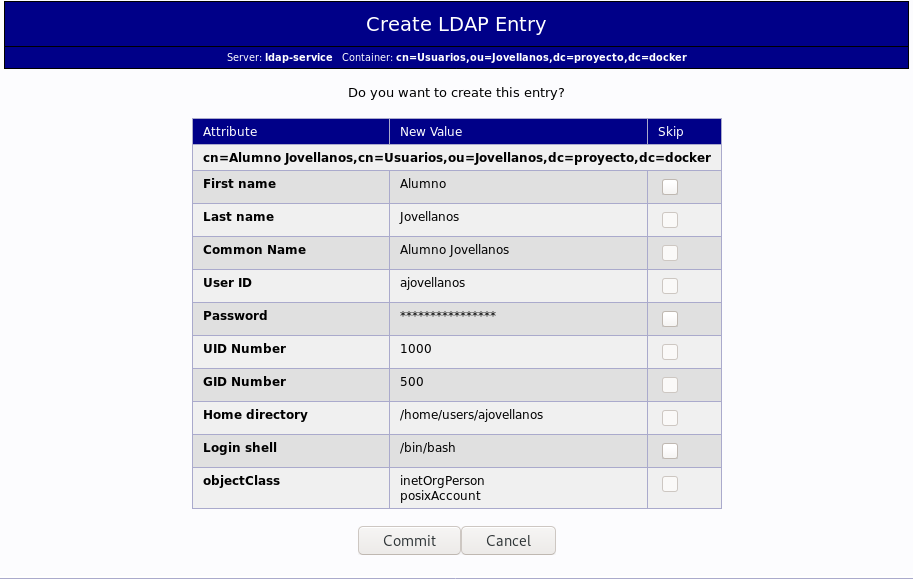


A continuación, creamos un usuario dentro del grupo usuario

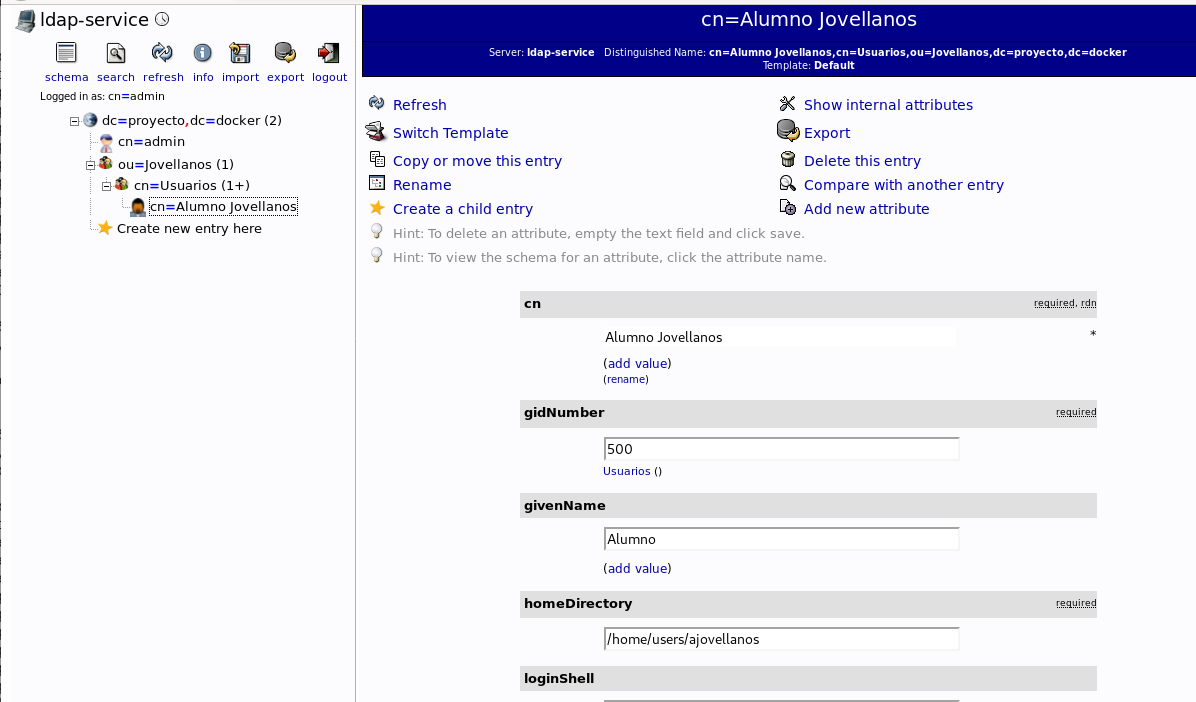




Como vemos podemos configurar múltiples parámetros como password, directorio home, la Shell que usara etc…



Y ya lo tendríamos creado

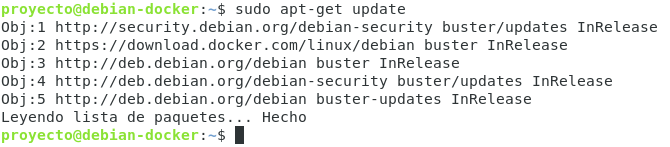


# Para justificar el usar Docker, hemos replicado el mismo proceso en el propio sistema Debian, sin usar Docker, para ver como Docker nos ayuda hacer despliegues mucho más rápido

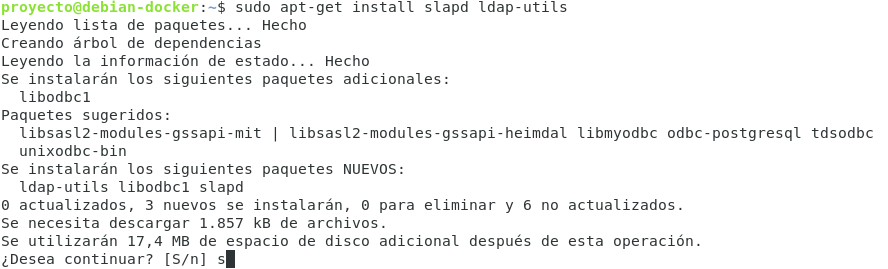
## Instalación OpenLdap Debian puro

Para empezar, realizaremos un

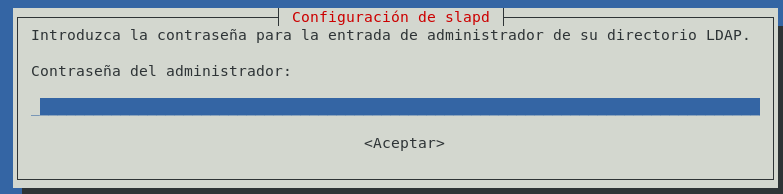
apt-get update



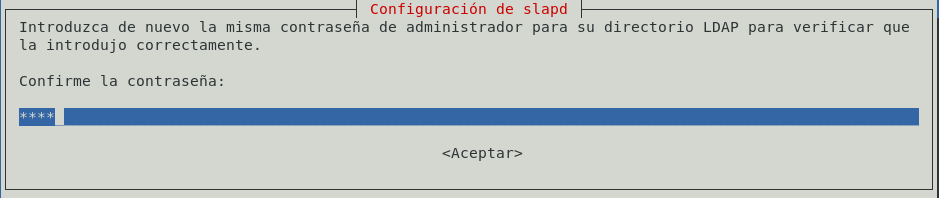
Una vez actualizado los repositorios nos descargaremos el paquete deseado



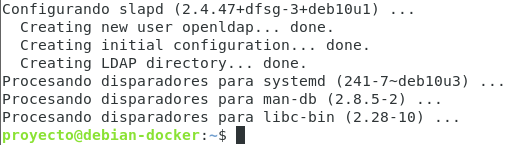
Una vez realizada la instalación del paquete nos pedirá la contraseña de administrador del directorio LDAP



Nos pide de nuevo una confirmación de contraseña

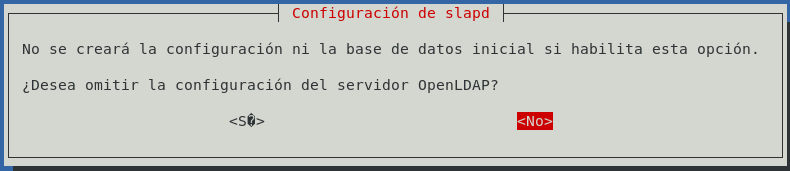


Tras esto, comprobamos que ya ha realizado por si solo una configuración inicial



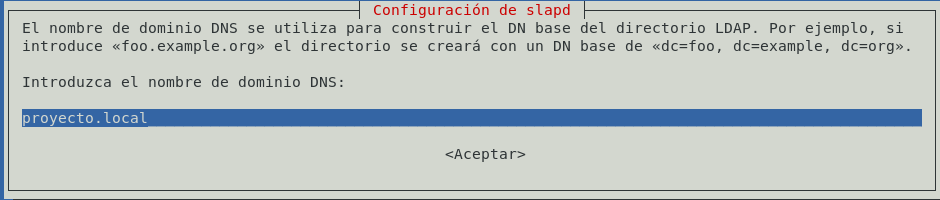
Como nosotros queremos una configuración propia ejecutaremos

dpkg-reconfigure -plow slapd

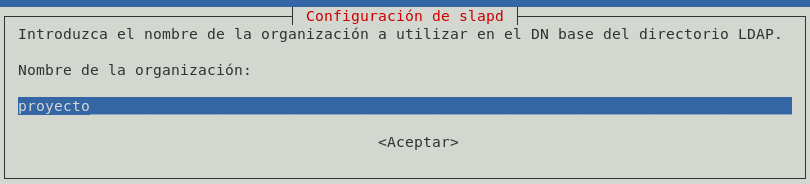


Así que aquí le daremos a la opción No

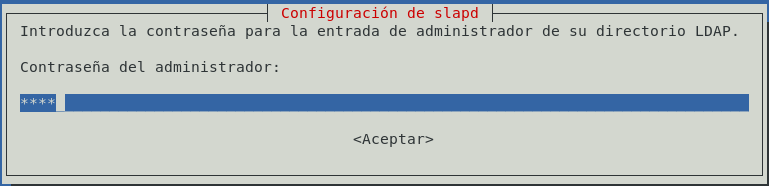
A continuación, nos pide que introduzcamos un nombre de dominio dns en nuestro caso será proyecto.Local



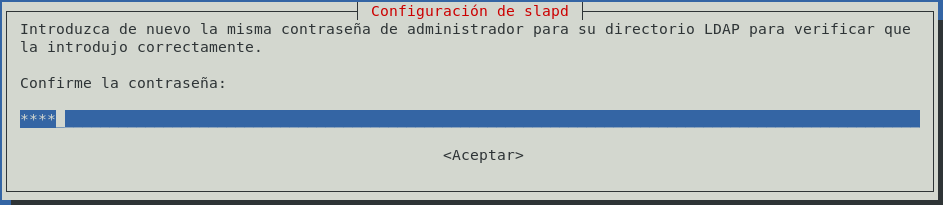
Nos pide un nombre de organización para utilizar en LDAP en nuestro caso será proyecto



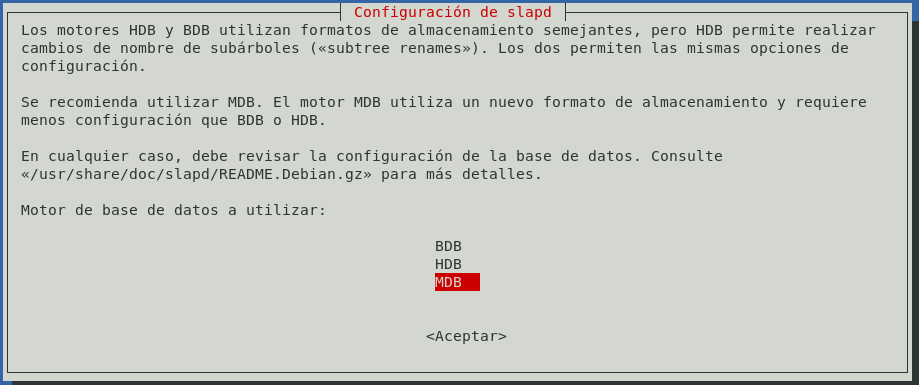
A continuación, nos pide la contraseña del administrador del directorio LDAP



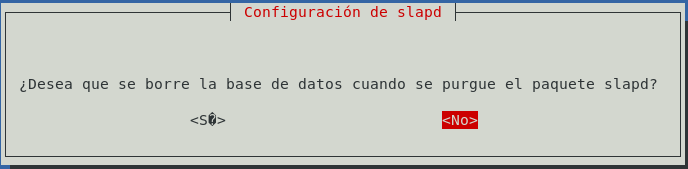
Como antes pide que la confirmemos introduciéndola de nuevo



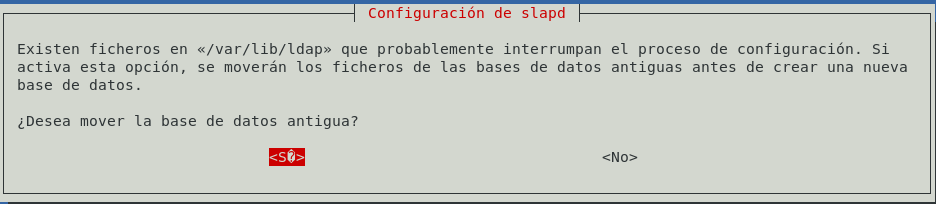
En el siguiente paso nos pide configurar el motor de base de datos que queremos usar, por defecto dejaremos el marcado MDB



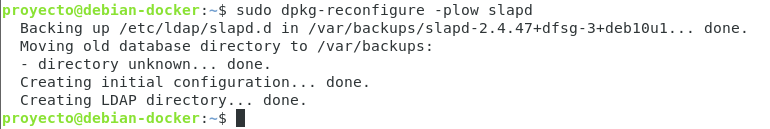
A continuación, nos pregunta que si deseamos que se elimine el directorio slapd de la base de datos cuando slapd sea eliminado, en su defecto dejamos No



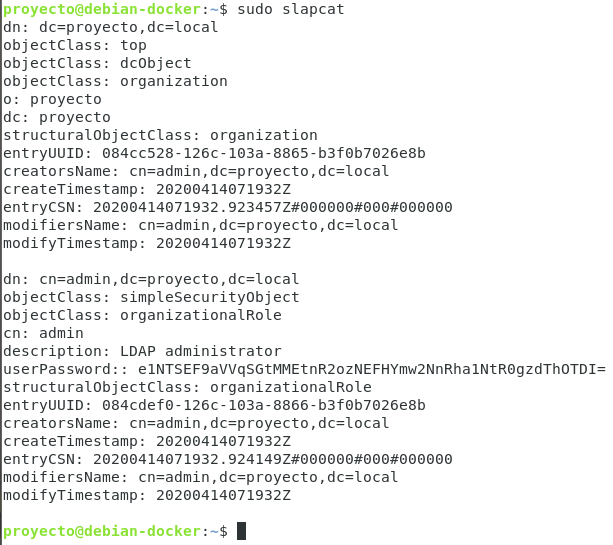
En la siguiente opción nos da a elegir si deseamos que se mueva el antiguo directorio, por defecto dejamos la opción Si



Una vez realizado el proceso nos ha vuelto a realizar la configuración, pero esta vez con nuestros parámetros



Para comprobar que todos los parámetros son correctos usaremos el comando slapcat que nos muestra el siguiente resultado

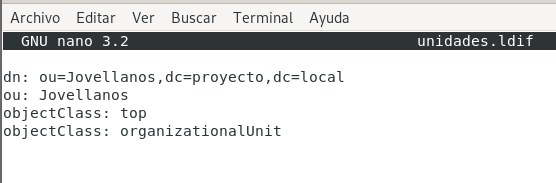


Con esto hemos finalizado la configuración inicial de LDAP, a continuación, pasaremos a crear los grupos, unidades organizativas y usuarios que deseemos

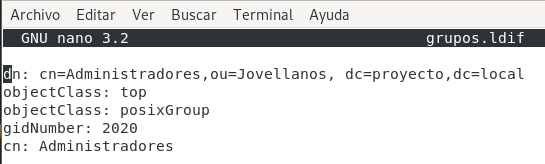
## Creación y configuración de Usuario, Grupos y Unidades Organizativas en LDAP en Debian puro

Para tener una organización fácil y optima vamos a manejar ficheros. ldif que a posterior se cargan en LDAPP a través de un comando, así a su vez podemos tener un fichero. ldif para poder separar usuarios, grupos y UO según nos venga en gana, A continuación, vamos a configurar los ficheros.

Primero crearemos la unidad organizativa Jovellanos

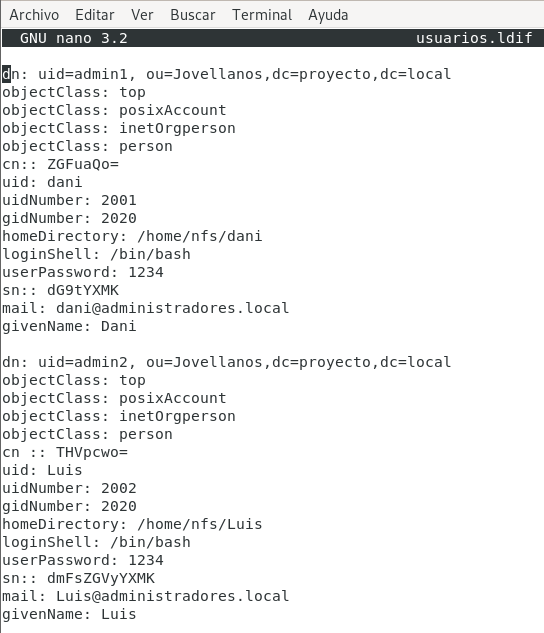


Ahora crearemos el grupo administradores



A continuación, hemos creado dos usuarios dentro de administradores, Tanto el cn como el sn que se requiere para este tipo de cuentas lo hemos realizado en base 64 con el comando

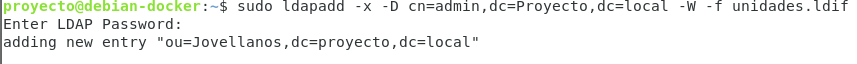
Echo dani | base64 y el resultado es el cn, repetimos el paso con el sn y en este caso ponemos el apellido, así hemos obtenido los valores requeridos



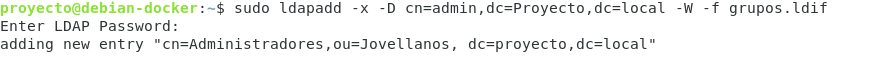
Ahora procederemos añadir a ldap los datos de cada fichero. ldif creado

ldapadd -x -D dc=Jovellanos,dc=local -W -f unidades.ldif

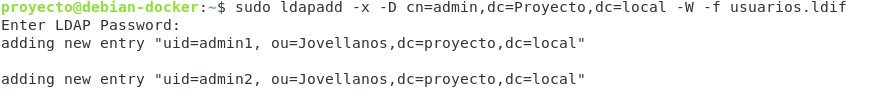
### Unidades organizativas



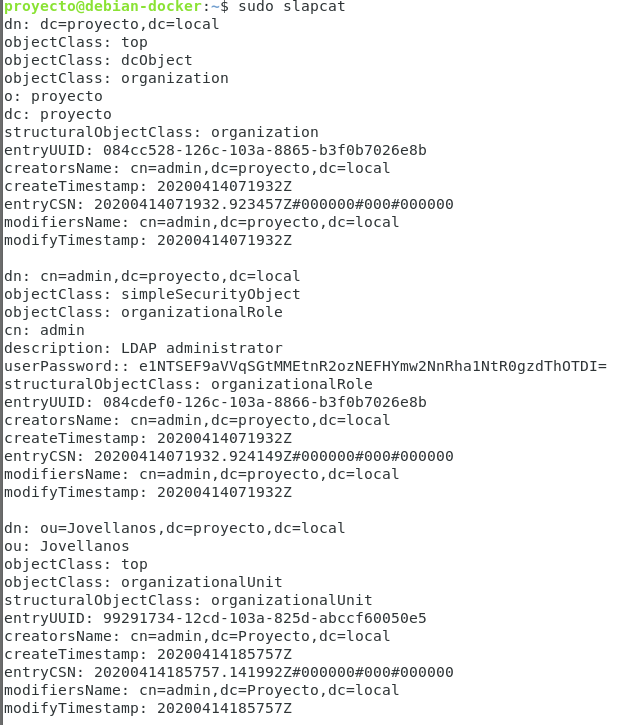
### Grupos

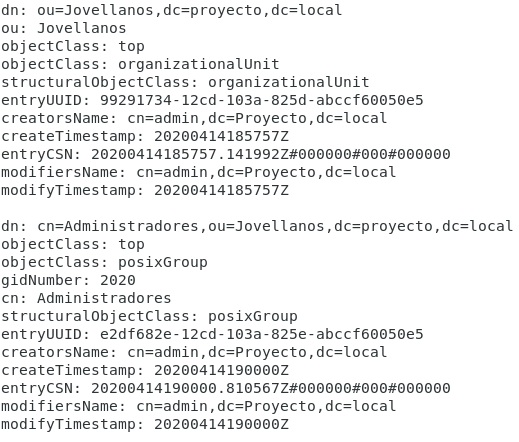


### Usuarios

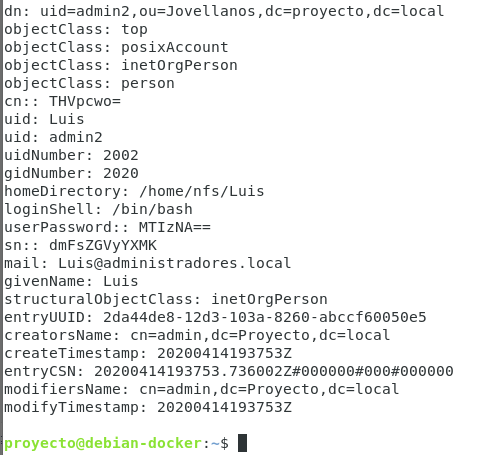


Tras estos procesos podemos usar el comando slapcat para comprobar la información







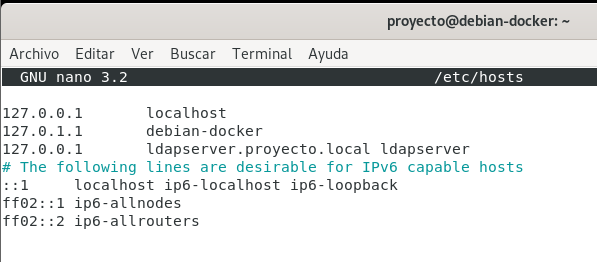


Con esto hemos finalizado la creación de UO, grupos y usuarios deseados para esta demo

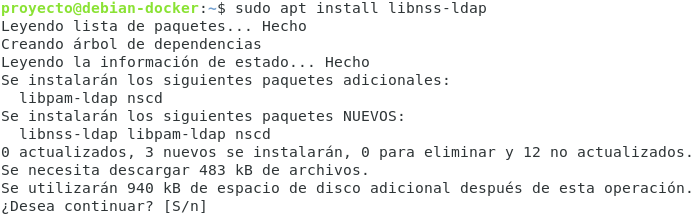
## Instalación De Interfaz Web Openldap en Debian puro

Para un manejo más fácil de Openldap instalaremos una interfaz web

Primero modificaremos el fichero /etc/hosts y añadimos la entrada del server

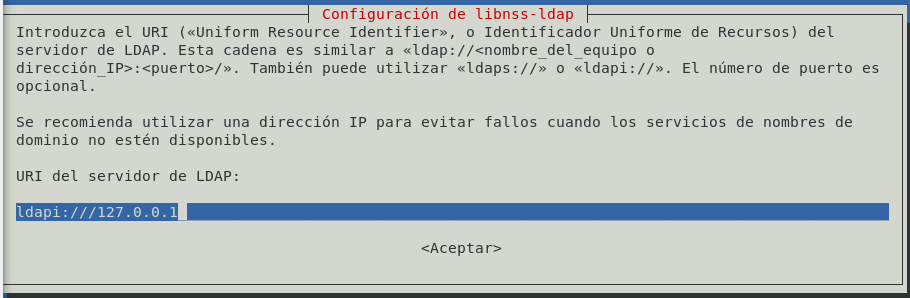


A continuación, procederemos a instalar el paquete libnss-ldap

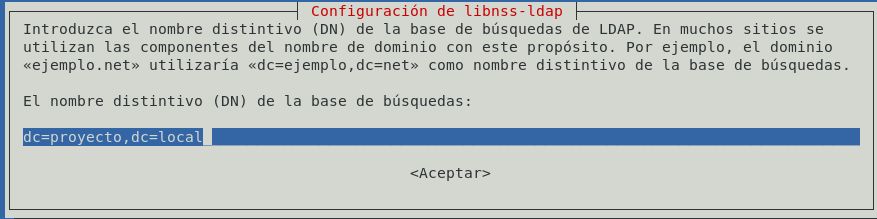


Proseguiremos con la configuración correspondiente

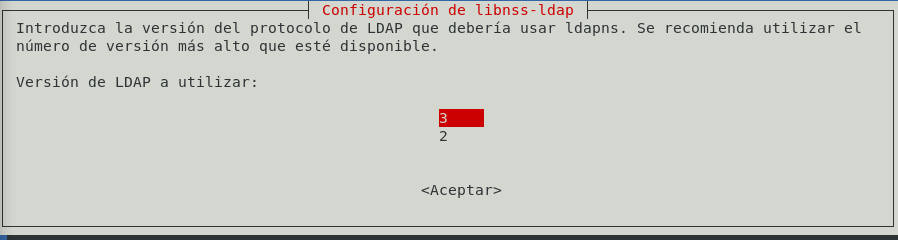
Aquí nos pide la ip que usaremos



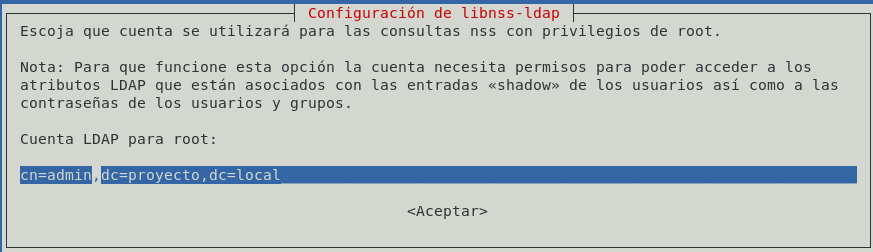
Aquí nos pide rellenar el nombre distintivo



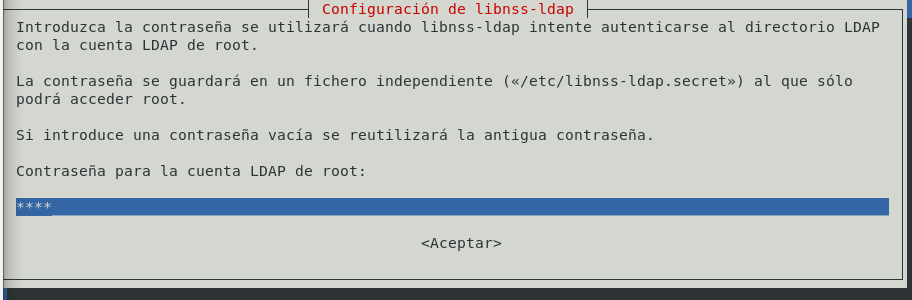
Aquí cogeremos el valor mas alto como indica



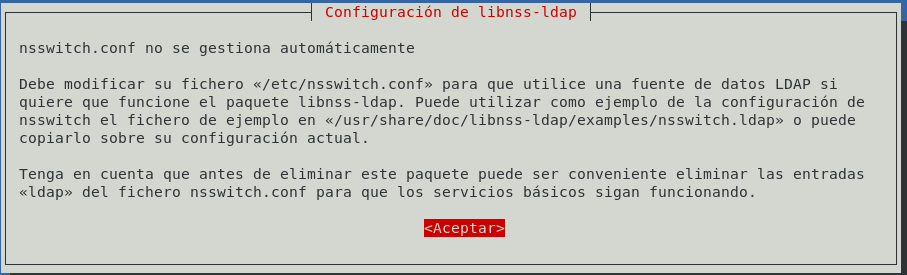
Configurando la cuenta admin para LDAP



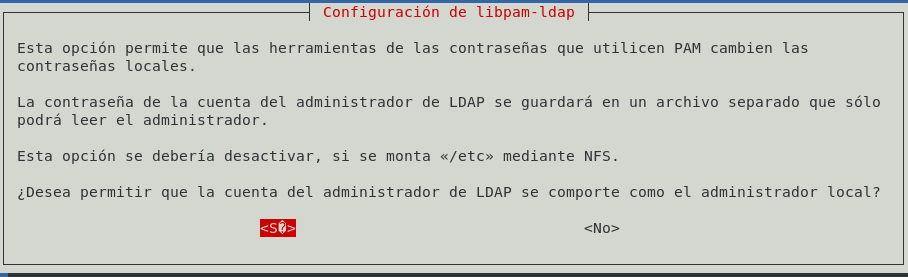
Contraseña admin

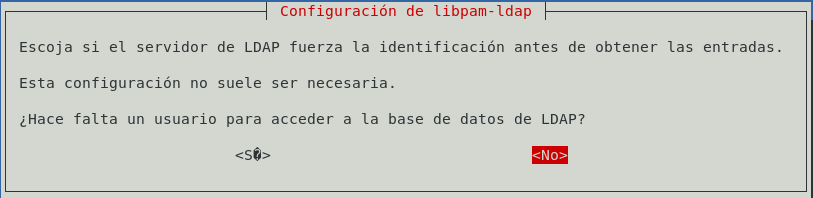


Aviso del fichero nsswitch.conf



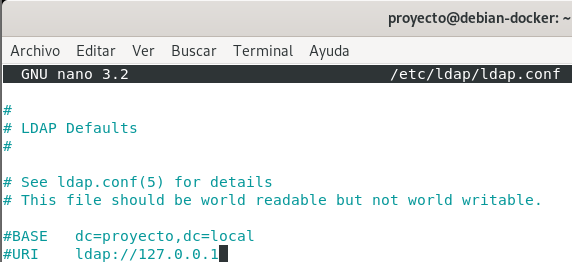
Aquí marcamos la opción por defecto





Vamos a configurar los clientes

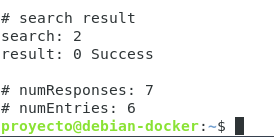




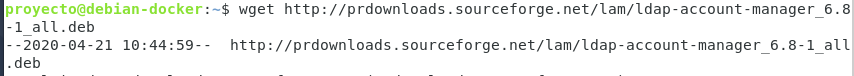
Ejecutaremos el siguiente comando para comprobar que funciona el servidor

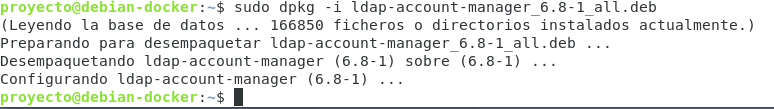


Nos devuelve un 0 así que es correcto



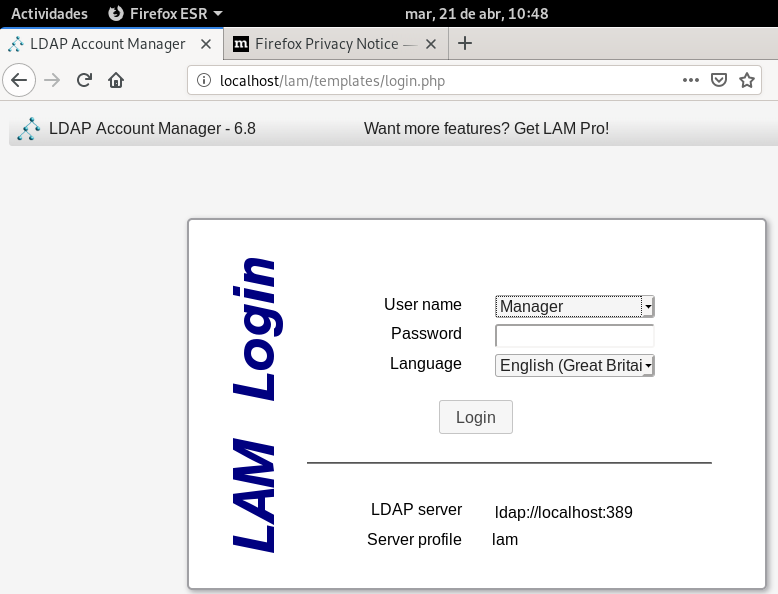
Terminamos bajando la interfaz web deseada



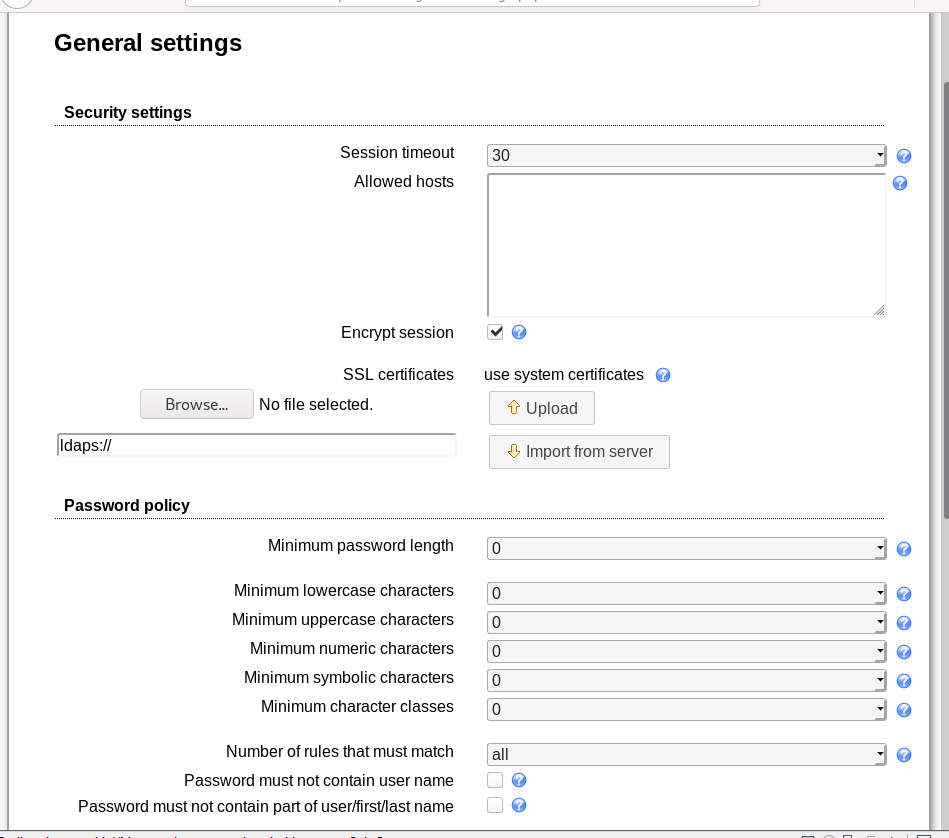


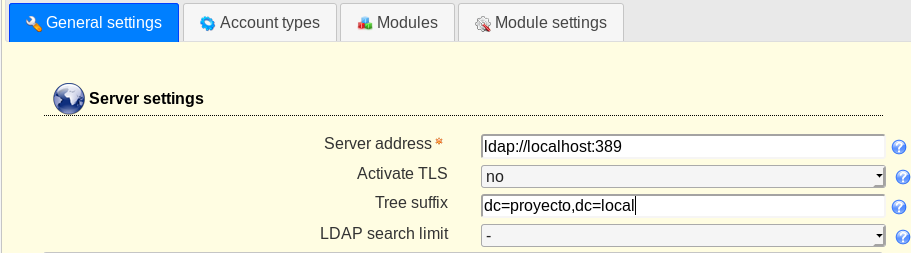
Comprobamos que tras esto podemos acceder

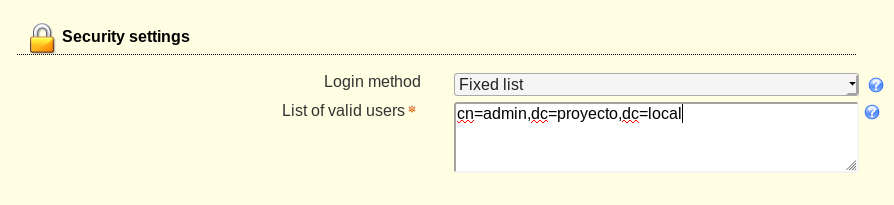
A la interfaz web



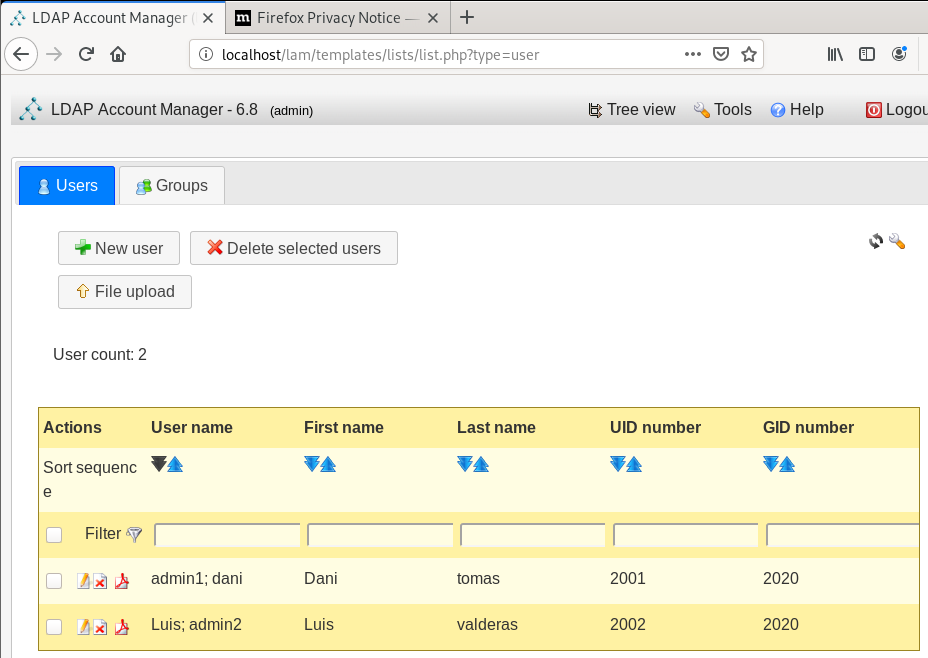
Vamos a proceder a configurarlo







Como vemos después de configurar los parámetros, nos reconoce los ficheros ldif creados



Y hasta aquí como se haría si no usásemos Docker, como vemos es un proceso más tedioso, pero con el mismo resultado (hemos usado otras interfaces web para comprobar e investigar la funcionalidad de cada una)