

Práctica 6. Almacenamiento sincronizado y compartido

Objetivo: En esta práctica evaluamos un recurso de almacenamiento cliente/servidor utilizado para sincronizar ficheros en red como es Nextcloud.

Presentar un documento pdf con la siguiente información:

-Capturas personalizadas de cada apartado donde se ve el tanto el servidor como los diferentes clientes de Nextcloud funcionando.

Opcional

-Combinar con ZeroTier para que servidores y clientes puedan estar ubicados en cualquier sitio.

Rúbrica:

Capturas de pantalla	30%
Descripción del trabajo desarrollado	20%
Entrega de apartados opcionales	20%
Entrega en plazo	30%

Desarrollo:

Se trata de instalar Nextcloud en nuestro ordenador como servidor utilizando contenedores Docker con docker-compose y lo comprobaremos con dos clientes, uno el propio ordenador y otro la máquina virtual con entorno gráfico de la práctica anterior.

1. Despliegue del servidor NextCloud

Siguiendo las indicaciones: <https://github.com/nextcloud/docker>

Utilizar el docker-compose y basándose en el ejemplo, desplegar el servidor version – apache con mariadb (mysql). No utilizar “Base version – FPM”.

El fichero docker-compose.yml que puede usar de base, es el siguiente aunque recuerde que:

-Debe fijar los valores a las variables MYSQL_ROOT_PASSWORD y MYSQL_PASSWORD con claves que considere “seguras”.

-Tener cuidado con los espacios delante en cada línea en los ficheros .yml , ya que definen el nivel de anidamiento.

```

services:
  db:
    image: mariadb:10.11
    restart: always
    command: --transaction-isolation=READ-COMMITTED --log-bin=binlog --binlog-format=ROW
    volumes:
      - db:/var/lib/mysql
    environment:
      - MYSQL_ROOT_PASSWORD=
      - MYSQL_PASSWORD=
      - MYSQL_DATABASE=nextcloud
      - MYSQL_USER=nextcloud

  redis:
    image: redis:alpine
    restart: always

  app:
    image: nextcloud
    restart: always
    ports:
      - 8080:80
    depends_on:
      - redis
      - db
    volumes:
      - nextcloud:/var/www/html
    environment:
      - MYSQL_PASSWORD=
      - MYSQL_DATABASE=nextcloud
      - MYSQL_USER=nextcloud
      - MYSQL_HOST=db

volumes:
  nextcloud:
  db:

```

Levantamos docker-compose

```
docker-compose up -d
```

En el puerto 8080 de nuestro ordenador tendremos instalado Nextcloud.

Accedemos a <http://localhost:8080> desde nuestro ordenador para continuar con la configuración, indicando que se utilizará la BBDD de mysql y las claves. Importante, como desplegamos desde docker-compose, el nombre del servidor de mysql es *db* (no localhost). Con docker-compose hay virtualmente dos nodos, uno el que ejecuta el nextcloud y el otro el que ejecuta la BBDD, por eso desde nextcloud cree” que la BBDD está en el nodo *db*

Una vez que estamos en el entorno de configuración podemos añadir usuarios.

2. Comprobación cliente Nextcloud.

Instalamos el cliente Nextcloud en nuestro ordenador.

<https://nextcloud.com/install/#install-clients>

E indicamos como dirección de nuestro servidor <http://localhost:8080>

Podemos utilizar nuestro nuevo usuario para crear y sincronizar nuestra carpeta nextcloud con nuestro servidor.

3. Comprobación cliente Nextcloud en la máquina vagrant con el escritorio de la práctica anterior.

Hay que indicar al Nextcloud que “confíe” en otros dominios ya que actúa como servidor en 192.168.56.1 para las máquinas Vagrant.

Paramos Nextcloud (con docker-compose down)

Modificamos el fichero config/config.php del contenedor de Nextcloud

```
'trusted_domains' =>
array (
  0 => '127.0.0.1:8080',
  1 => '192.168.56.1:8080',
),
```

Reiniciamos Nextcloud (con docker-compose up -d)

Instalamos el cliente Nextcloud en la máquina virtual con el escritorio.

<https://nextcloud.com/install/#install-clients>

E indicamos como dirección de nuestro servidor <http://192.168.56.1:8080>

4. Apartado opcional. Combinando la red ZeroTier que ha creado en la práctica anterior, modificar el fichero config/config.php para que también acepte la IP asignada al servidor.

Realice alguna captura donde se observe que el cliente está utilizando la IP de la red de ZeroTier.

Anexo: Cómo utilizar Nextcloud con Raspberry

El siguiente fichero docker-compose.yml es un ejemplo de cómo utilizar Nextcloud en una Raspberry.

```
version: '2'

services:
  db:
    image: yobasystems/alpine-mariadb:10.5.11-arm32v7
    command: --transaction-isolation=READ-COMMITTED --binlog-format=ROW
    restart: "no"
    volumes:
      - /media/btrfs1/nextcloud/db:/var/lib/mysql
    environment:
      - MYSQL_ROOT_PASSWORD=PAssWoRD%RoOT
      - MYSQL_PASSWORD=PaSsWoRd%BaSeDaTos
      - MYSQL_DATABASE=nextcloud
      - MYSQL_USER=nextcloud

  app:
    image: arm32v7/nextcloud
    ports:
      - 8080:80
    links:
      - db
    volumes:
      - /media/btrfs1/nextcloud/html:/var/www/html
    restart: "no"
```

Donde la carpeta */media/btrfs1/nextcloud* podría ser una carpeta en nuestro disco duro externo en la Raspberry, de forma que todo el almacenamiento queda alojado en dicho disco de mayor capacidad.