

Práctica Docker 3. DockerFile. Docker Compose

Dockerfile

1. Crea unha carpeta servidores, na que teñas 2 carpetas: unha ubuntu, e outra alpine. Gardarás en cada unha delas os DockerFile correspondentes
2. Comproba que funciona o exemplo dos apuntes no que se crea unha imaxe desde ubuntu, instalando apache. Crea unha imaxe propia a partir de ubuntu, na que teñas instalado apache
3. Crea unha imaxe similar á anterior, empregando como base alpine. O equivalente a apt en AlpineLinux é **apk**, co que temos que facer algúns cambios:
 - a) para actualizar paquetes debes usar **apk update**, en vez de **apt update**.
 - b) para instalar paquetes debes usar **apk add**, en vez de **apt install** (non se permite o modificador -y). O apache ten o mesmo nome apache2
 - c) O servizo de apache chámase "**httpd**" (atópase en "**/usr/bin/httpd**")
4. Canto ocupan as imaxes creadas?
5. Crea un contedor coas imaxes anteriores, un servidor escoitando en 8080, e outro en 8081. Comproba que te podes conectar desde un navegador a ambos servidores. Finalmente, páraos, e elimina os 2 contedores se non se eliminaron ao paralos

Docker Compose

Imos facer o mesmo que fixemos co anterior: usaremos as imaxes creadas co **Dockerfile**, empregando Docker Compose

6. Na carpeta de *ubuntu* fai un **compose.yml** para iniciar o servidor de *ubuntu*, expoñendo o porto 8080, e montando un volume local **www**, na carpeta do contedor **/var/www/html** . Comproba que funciona e que te podes conectar desde o navegador
7. Fai o mesmo desde a carpeta alpine, cun **compose.yml**. Agora expoñerá o porto 8081, e a carpeta na se gardarán os contidos será no contedor **/usr/local/apache2/htdocs**
8. Para ambos contedores, e comproba que os contedores se eliminaron
9. Finalmente, crea unha carpeta servidores, na que existirá 2 carpetas: **ubuntu** e **alpine**, cada unha cunha carpeta **www**. Crea na carpeta Servidores un único **compose.yml** que levante os 2 servidores, coa mesma configuración que nos apartados anteriores.