

Puesta en Producción Segura

Práctica 2: DevSecOps

Paso 1: Imagen PHP Básica creada.....	3
PASO 2: Imagen Ubuntu + Apache + PHP descargando app.....	4
PASO 3: Crear y probar contenedores.....	5
PASO 4: Lanzar 20 contenedores en puertos distintos.....	6

Paso 1: Imagen PHP Básica creada

Primero creamos la carpeta donde irá ésta app, y luego creamos en ésta el index.php y el Dockerfile.

```
usuario@pps: ~/diegogf-practica-devsecops

usuario@pps:~$ mkdir -p diegogf-practica-devsecops/php-app
usuario@pps:~$ cd diegogf-practica-devsecops/php-app/
usuario@pps:~/diegogf-practica-devsecops/php-app$ nano index.php
usuario@pps:~/diegogf-practica-devsecops/php-app$ nano Dockerfile

GNU nano 4.8                                php-app/index.php
?php
echo "¡Hola Mundo desde PHP en Docker!";
?>

GNU nano 4.8                                php-app/Dockerfile
Imagen oficial PHP
FROM php:8.1-cli

# Carpeta de trabajo
WORKDIR /var/www/html

# Copiar app desde el host
COPY . /var/www/html

# Exponer puerto 80
EXPOSE 80

# Ejecutar servidor embebido
CMD ["php", "-S", "0.0.0.0:80", "-t", "/var/www/html"]
```

Subimos lo hecho al git-hub, primero iniciando un git, vinculando a un proyecto ya existente tuyo en github, y luego hacemos el commit seguido del push para añadirlo al proyecto en github dando error éste.

```
usuario@pps:~/diegogf-practica-devsecops/php-app$ cd ../
usuario@pps:~/diegogf-practica-devsecops$ git init
Inicializado repositorio Git vacío en /home/usuario/diegogf-practica-devsecops/.git/
usuario@pps:~/diegogf-practica-devsecops$ git remote add origin https://github.com/DiegoGFCiber/diegogf-practica-devsecops.git
usuario@pps:~/diegogf-practica-devsecops$ git add php-app/
usuario@pps:~/diegogf-practica-devsecops$ git commit -m "Paso 1: Imagen PHP básica creada"
[master (commit-raiz) b4f88df] Paso 1: Imagen PHP básica creada
 2 files changed, 17 insertions(+)
 create mode 100644 php-app/Dockerfile
 create mode 100644 php-app/index.php
usuario@pps:~/diegogf-practica-devsecops$ git push origin main
error: src refspec main no concuerda con ninguno
error: falló el empuje de algunas referencias a 'https://github.com/DiegoGFCiber/diegogf-practica-devsecops.git'
```

El error se arregla porque de primeras es como que no estaba apuntando hacia la rama main, entonces nos movemos a ella y luego repetimos el push, y vemos como funciona.

```
usuario@pps:~/diegogf-practica-devsecops$ git checkout -b main
Cambiado a nueva rama 'main'
usuario@pps:~/diegogf-practica-devsecops$ git push -u origin main
Username for 'https://github.com': DiegoGFCiber
Password for 'https://DiegoGFCiber@github.com':
Enumerando objetos: 5, listo.
Contando objetos: 100% (5/5), listo.
Compresión delta usando hasta 10 hilos
Comprimiendo objetos: 100% (4/4), listo.
Escribiendo objetos: 100% (5/5), 559 bytes | 186.00 KiB/s, listo.
Total 5 (delta 0), reusado 0 (delta 0)
To https://github.com/DiegoGFCiber/diegogf-practica-devsecops.git
* [new branch]      main -> main
Rama 'main' configurada para hacer seguimiento a la rama remota 'main' de 'origin'.
```

PASO 2: Imagen Ubuntu + Apache + PHP descargando app

Como antes empezamos creando una carpeta donde crearemos en este caso solo el Dockerfile, porque la aplicación la cogeremos de internet.

```
usuario@pps:~/diegogf-practica-devsecops$ mkdir ubuntu-apache-php
usuario@pps:~/diegogf-practica-devsecops$ cd ubuntu-apache-php/
usuario@pps:~/diegogf-practica-devsecops/ubuntu-apache-php$ nano Dockerfile
```

```
GNU nano 4.8          ubuntu-apache-php/Dockerfile
# Imagen base Ubuntu
FROM ubuntu:22.04

# Evitar interacción al instalar paquetes
ENV DEBIAN_FRONTEND=noninteractive

# Actualizar e instalar Apache, PHP, Git
RUN apt-get update && apt-get install -y apache2 php libapache2-mod-php git curl

# Por si acaso limpiamos posibles restos en /var/www/html
RUN rm -rf /var/www/html

# Descargar app PHP sencilla
RUN git clone https://github.com/heroku/php-getting-started.git /var/www/html

# Dar permisos correctos
RUN chown -R www-data:www-data /var/www/html

# Exponer puerto 80
EXPOSE 80

# Ejecutar Apache en primer plano
CMD ["apachectl", "-D", "FOREGROUND"]
```

Y hacemos otra vez el commit conforme añadimos el nuevo contenido

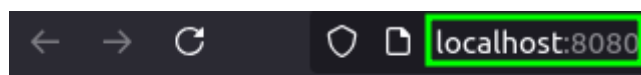
```
usuario@pps:~/diegogf-practica-devsecops/ubuntu-apache-php$ cd ..
usuario@pps:~/diegogf-practica-devsecops$ git add ubuntu-apache-php/
usuario@pps:~/diegogf-practica-devsecops$ git commit -m "Paso 2: Imagen Ubuntu + Apache + PHP creada"
[master 561b883] Paso 2: Imagen Ubuntu + Apache + PHP creada
1 file changed, 23 insertions(+)
 create mode 100644 ubuntu-apache-php/Dockerfile
usuario@pps:~/diegogf-practica-devsecops$ git push origin main
Username for 'https://github.com': DiegoGFCiber
Password for 'https://DiegoGFCiber@github.com':
Enumerando objetos: 5, listo.
Contando objetos: 100% (5/5), listo.
Compresión delta usando hasta 10 hilos
Comprimiendo objetos: 100% (3/3), listo.
Escribiendo objetos: 100% (4/4), 750 bytes | 750.00 KiB/s, listo.
Total 4 (delta 0), reusado 0 (delta 0)
To https://github.com/DiegoGFCiber/diegogf-practica-devsecops.git
 b4f88df..561b883  main -> main
```

PASO 3: Crear y probar contenedores

Ahora creamos los dockers, los arrancamos y comprobamos si funcionan.

Primero lo hacemos con el que usa una aplicación de un directorio local, index.php en este caso.

```
usuario@pps:~/diegogf-practica-devsecops$ cd php-app/
usuario@pps:~/diegogf-practica-devsecops/php-app$ docker build -t diegogfapp-php .
[+] Building 34.5s (8/8) FINISHED                                docker:default
=> [internal] load build definition from Dockerfile
0.0s
=> => transferring dockerfile: 286B
0.0s
=> [internal] load metadata for docker.io/library/php:8.1-cli
2.7s
usuario@pps:~/diegogf-practica-devsecops/php-app$ docker run -d -p 8080:80 --name contenedor-php diegogfapp-php
f4142d620a0bf879fe5bbc67b1d00e1ce7513271e35460744c9e6d4c18705b81
```



¡Hola Mundo desde PHP en Docker!

Entonces hacemos lo mismo ahora para el docker que usa una aplicación en la red.

```
usuario@pps:~/diegogf-practica-devsecops/php-app$ cd ../ubuntu-apache-php/
usuario@pps:~/diegogf-practica-devsecops/ubuntu-apache-php$ docker build -t diegogfapp-apache-php .
ERROR: "docker buildx build" requires exactly 1 argument.
See 'docker buildx build --help'.

Usage:  docker buildx build [OPTIONS] PATH | URL | -

Start a build
usuario@pps:~/diegogf-practica-devsecops/ubuntu-apache-php$ docker build -t diegogfapp-apache-php .
[+] Building 29.1s (9/9) FINISHED                                docker:default
=> [internal] load build definition from Dockerfile
0.0s
=> => transferring dockerfile: 648B
0.0s
=> [internal] load metadata for docker.io/library/ubuntu:22.04
usuario@pps:~/diegogf-practica-devsecops/ubuntu-apache-php$ docker run -d -p 8081:80 --name contenedor-apache-php diegogfapp-apache-php
a49d9e475b752d97283329c1fabes7e570161064b4e1bd6b8d5174cf3fffee13
```

←

→

↻

🛡️

📄

localhost:8081

Index of /

	Name	Last modified	Size	Description
🔍	Procfile	2025-05-07 15:29	29	
📄	README.md	2025-05-07 15:29	2.4K	
🔍	app.json	2025-05-07 15:29	199	
🔍	composer.json	2025-05-07 15:29	308	
🔍	composer.lock	2025-05-07 15:29	57K	
📁	views/	2025-05-07 15:29	-	
📁	web/	2025-05-07 15:29	-	

Apache/2.4.52 (Ubuntu) Server at localhost Port 8081

PASO 4: Lanzar 20 contenedores en puertos distintos

Creo un script que al ejecutarlo crea 20 dockers, y otro que los parará y eliminará. Y le damos los permisos necesarios para poder ejecutarlos.

```

usuario@pps:~/diegogf-practica-devsecops/ubuntu-apache-php$ cd ..
usuario@pps:~/diegogf-practica-devsecops$ nano Arrancar20Contenedores.sh
usuario@pps:~/diegogf-practica-devsecops$ sudo chmod 755 Arrancar20Contenedores.sh
[sudo] contraseña para usuario:
usuario@pps:~/diegogf-practica-devsecops$ ./Arrancar20Contenedores.sh
91d2acb9b30c5399dac5fd2c8b50689fb1471ab9ab05f77acc2e611541299e9c
f8d7672529e4c67c6613928d9d132b055dd204247f6c828ff14b297b36e59323
5f5f84ddf8b067b8678ca6ee31fc36fb4e6f206ab2625ef0e26a511560b2eae2
e86b63a61a56b0d05f8fee4e45fe33461e52044839e55355f6a1160f189af9ac
642ec67cf415d3b9aa24f037b6a84f73f81ebd1bc88bb2e06143462823da41c7
0f51e744596838562766e48f61bf393c55f0354568b03143064b111c826d46c3
3efec62c4f64805b0b70c52e0acdb0731c631f1722bf9ac74e3d2594dfd8e315
3001ee01488b51486f2d050df766fd05843fed4dceca15f83783880fba3f3a77
ba2529ab821f54c583997bfbcb88b56cf19cd300639535d174923c35b72340207
8603dffc2933ccf0761bdc8b7a8d113e9e930dc4f818810ea55110bfd019c8f0
1ec2d7c7fd739e9669d8e88cdd7842920951267fe310d024b707773e9b346131
3c68610f5026a476c2d6c42b79f0477c30c5a8f877e2a207dcbfa9720de70c9d
8992342d2a5368ef8b393c6b7daa6af979eb7acdce27daa2096fb093f7451bc6
19df606ba1b29b028e07f77998b7f2bedf1b847ace59c92ab706dbe928b66e92
9178cbe5d71edd334e040e88c794c53bdd1616ce7f5704b090898abd043d6b56
549ad5fb0f7cd8572baf9149cc090fbcf5dd7d60fc11f501c6e80360530c0bc2
84291c7b7554affb688d7e21e7893efb3633247281383d352ebb466fbb0566b8
02180ea14553e82ae4d1da2fd7da69b1b695f5864b5db082539412a467d35c3b
46e2d46ff43303a7da2401b37514dfa9fdd3f0890a1def0de6e284345d3d2b39
ce3082660e85ffdf8af5aebb73f76228fe7082f84ab2026bfff495d7560396d6
usuario@pps:~/diegogf-practica-devsecops$ nano Detener20Contenedores.sh
usuario@pps:~/diegogf-practica-devsecops$ sudo chmod 755 Detener20Contenedores.sh
usuario@pps:~/diegogf-practica-devsecops$ ./Detener20Contenedores.sh
ce3082660e85
46e2d46ff433
02180ea14553
84291c7b7554
549ad5fb0f7c
9178cbe5d71e
19df606ba1b2
8992342d2a53
3c68610f5026
1ec2d7c7fd73
8603dffc2933
ba2529ab821f
3001ee01488b
3efec62c4f64
0f51e7445968
642ec67cf415
e86b63a61a56
5f5f84ddf8b0
f8d7672529e4
91d2acb9b30c
ce3082660e85

```

```

usuario@pps: ~/diegogf-practica-devsecops
GNU nano 4.8 Arrancar20Contenedores.sh
for i in {1..20}; do
  docker run -d -p $((8000+i)):80 --name apache-php-$i diegogfapp-apache-php
done

```

```

usuario@pps: ~/diegogf-practica-devsecops
GNU nano 4.8 Detener20Contenedores.sh
docker stop $(docker ps -q --filter "name=apache-php-")
docker rm $(docker ps -aq --filter "name=apache-php-")

```

E hice un añadido final al git , con un README y avisando conforme los contenedores han sido probados y funcionan.

```
usuario@pps:~/diegogf-practica-devsecops$ nano RADME.md
usuario@pps:~/diegogf-practica-devsecops$ mv RADME.md README.md
usuario@pps:~/diegogf-practica-devsecops$ git add README.md
usuario@pps:~/diegogf-practica-devsecops$ git commit -m "Paso 3: Contenedores probados y memoria creada"
[main a71a658] Paso 3: Contenedores probados y memoria creada
 1 file changed, 1 insertion(+)
 create mode 100644 README.md
usuario@pps:~/diegogf-practica-devsecops$ git push origin main
Username for 'https://github.com': DiegoGFCiber
Password for 'https://DiegoGFCiber@github.com':
Enumerando objetos: 4, listo.
Contando objetos: 100% (4/4), listo.
Compresión delta usando hasta 10 hilos
Comprimiendo objetos: 100% (2/2), listo.
Escribiendo objetos: 100% (3/3), 352 bytes | 352.00 KiB/s, listo.
Total 3 (delta 0), reusado 0 (delta 0)
To https://github.com/DiegoGFCiber/diegogf-practica-devsecops.git
 561b883..a71a658  main -> main
```