

# Ambiente de Computación Distribuido

## Resumen

En este documento se describen los objetivos y las instrucciones para lograr el uso de herramientas de un ambiente de computación distribuida. Además, se presentan las condiciones de entrega del informe, contenido y formato del informe.

## Tabla de contenido

|  |          |
|--|----------|
| <b>Objetivos</b>                           | <b>2</b> |
| <b>Descripción del ejercicio</b>           | <b>2</b> |
| Proxy inverso                              | 3        |
| ¿Qué debe hacer el/la estudiante?          | 4        |
| Docker-compose                             | 4        |
| Demo en Google Cloud Console               | 4        |
| Servidor web de Apache                     | 6        |
| <b>Algunos comentarios</b>                 | <b>7</b> |
| Respecto a la ejecución del ejercicio      | 7        |
| Respecto a la conectividad de las máquinas | 7        |
| <b>Entregables</b>                         | <b>7</b> |
| <b>Número de integrantes</b>               | <b>8</b> |
| <b>Fecha de entrega</b>                    | <b>8</b> |

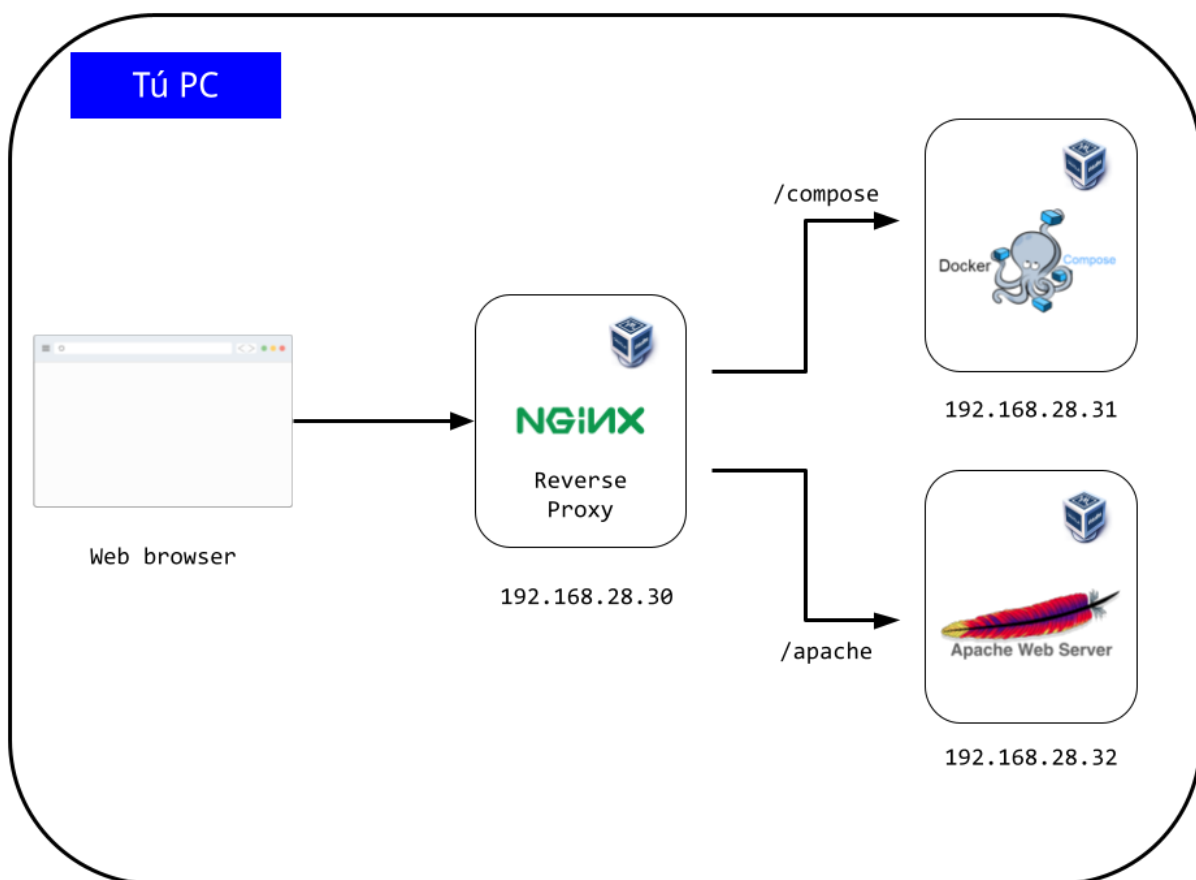
# Objetivos

A continuación se enumeran algunos de los objetivos que el estudiante deberá alcanzar y posteriormente evidenciar a través del informe final y la sustentación del ejercicio.

- El estudiante debe demostrar capacidad en la utilización de herramientas de virtualización, e.g. VirtualBox o KVM/QEMU.
- El estudiante debe ser capaz de instalar un sistema operativo tipo Unix/Linux en una máquina virtual.
- El estudiante estará en la capacidad de instalar y poner a punto [al menos] un servicio de red.
- El estudiante estará en la capacidad de hacer el despliegue de una aplicación que se ejecute sobre un ambiente de computación distribuida.

## Descripción del ejercicio

En este ejercicio, el/la estudiante llevará a cabo el despliegue de un ambiente computacional distribuido. En este caso el/la estudiante hará del despliegue de los servicios representados en la figura de abajo.



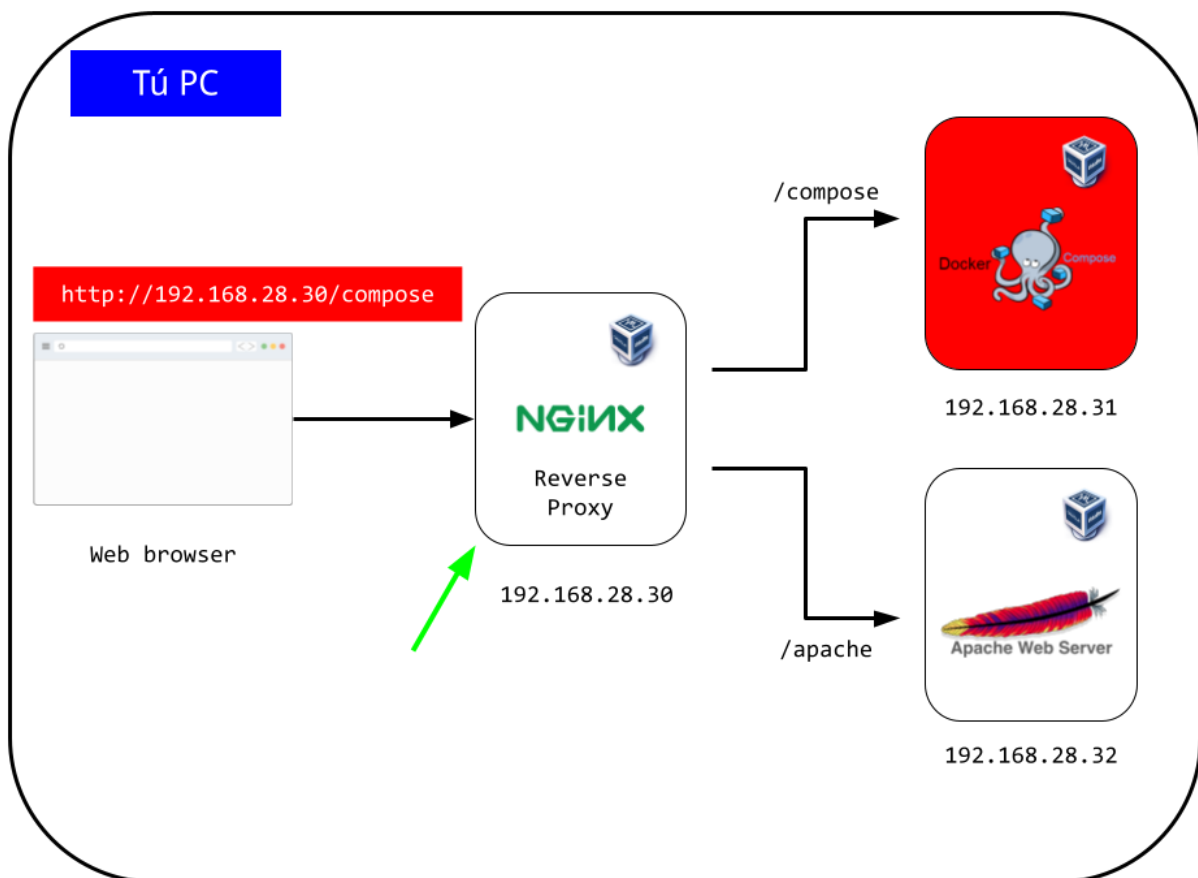
Esta figura ilustra al menos 5 elementos:

- **Tú PC**
- Un web browser o navegador
- Un proxy inverso (IP: 192.168.28.30 y ejecutándose sobre una máquina virtual)
- Un aplicativo ejecutado con la herramienta 'docker-compose' (IP: 192.168.28.31 y ejecutándose sobre una máquina virtual)
- Un servidor web Apache (IP: 192.168.28.32 y ejecutándose sobre una máquina virtual)

Usted ya cuenta con dos elementos de los mencionados anteriormente, **Tú PC** y el web browser y también ya sabe crear máquinas virtuales. Entonces, usted se debe encargar de hacer el despliegue (instalación y configuración) de los otros elementos en máquinas virtuales a saber: proxy inverso, máquina virtual ejecutando 'docker-compose' y servidor web Apache.

## Proxy inverso

Un [proxy inverso](#) es un servicio de red encargado de recibir peticiones de red y redireccionar dichas peticiones a un servicio o recurso computacional habilitado para procesar ese tipo de peticiones. En la siguiente figura podemos ver el proxy inverso en acción.



En este caso el proxy inverso (192.168.28.30) se encargará de recibir peticiones HTTP y delegar el procesamiento de dichas peticiones a algunas de las máquinas virtuales (ó 192.168.28.31 ó 192.168.28.32).

En el ejemplo de la gráfica anterior, desde un 'web browser' se hace el acceso al URL '<http://192.168.28.30/compose>'. Esta petición es recibida por el 'reverse proxy' (192.168.28.30) y este se encarga de delegar el procesamiento de dicha petición a la máquina cuyo IP es 192.168.28.31.

Si se envía en su lugar la petición '<http://192.168.28.30/apache>' a quien delegará el 'proxy inverso' la atención de la petición es a la máquina virtual con IP 192.168.28.32.

## ¿Qué debe hacer el/la estudiante?

De lo que se debe encargar el/la estudiante es de llevar a cabo el proceso de instalación y configuración del proxy inverso (o reverse proxy) a través de la herramienta Nginx en su máquina virtual.

## Docker-compose

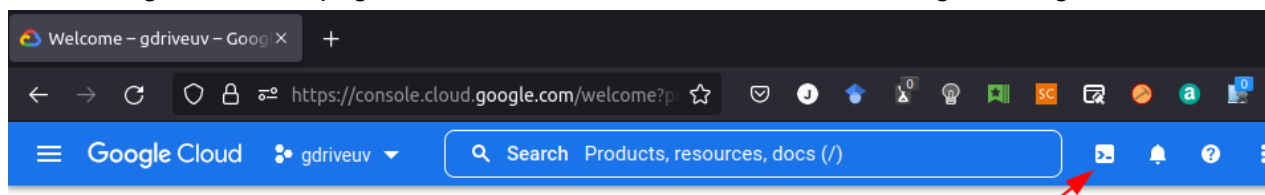
El/la estudiante debe preparar una máquina virtual en la cual se llevará a cabo la ejecución de una aplicación basada en contenedores de Docker usando la herramienta 'docker-compose'. La aplicación será provista por el instructor y de lo que se debe encargar el/la estudiante es de instalar la herramienta 'docker-compose' en la máquina virtual y de ejecutar con esta herramienta la aplicación provista por el instructor.

A continuación se hace una descripción de los pasos más relevantes para ejecutar la aplicación mencionada en el párrafo anterior a modo de demostración en un entorno de nube pública.

## Demo en Google Cloud Console<sup>1</sup>

Para llevar a cabo la ejecución de la aplicación debe ingresar al siguiente enlace <https://console.cloud.google.com/>. Para la ejecución de este demo ó puede usar su cuenta de correo institucional o la personal.

Una vez ingresa en esa página de click sobre el ícono señalado en la siguiente figura:



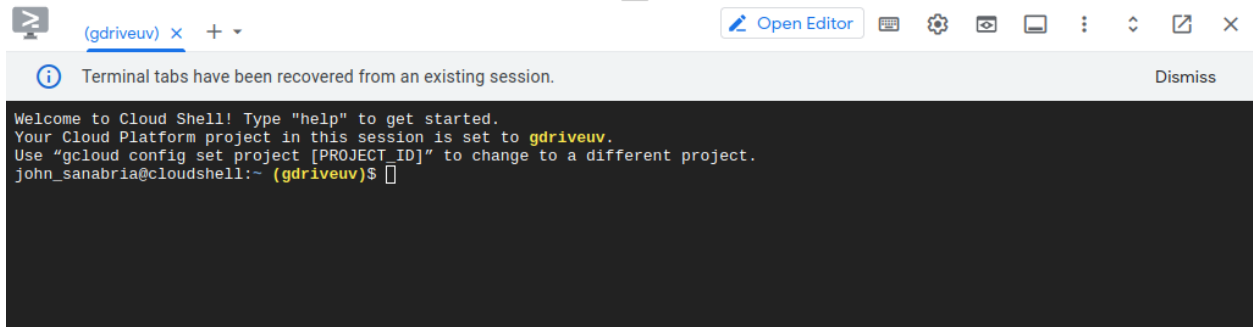
Esto le deberá abrir, en la parte inferior de la página web la siguiente interfaz

---

<sup>1</sup> Este demo se llevó a cabo el 3 de agosto del 2022.

## Quick access

[Privacy Policy](#) · [Terms of Service](#)



En este momento usted ya tiene una ‘máquina virtual’ en la nube que le permitirá ejecutar ciertas tareas. Google Cloud Console le permite usar esta ‘máquina virtual’ a través de una terminal.

Para ejecutar la aplicación con ‘docker-compose’ ubíquese en la terminal provista por Google Cloud Console y ejecute los siguientes comandos:

```
export CLOUDSDK_PYTHON=/usr/bin/python2
```

A continuación se hará la descarga del aplicativo que se ejecutará con la herramienta ‘docker-compose’. Para ello cree un archivo llamado ‘script.sh’. Ejecute el siguiente comando:

```
nano script.sh
```

Y copie las siguientes líneas dentro del editor:

```
#!/usr/bin/env bash
CURRENT_WORKDIR=$(pwd)
FILE_ID="1EH1zS1wQ_5WmvUEar2eXcB4QMj1Mo49g"
FILENAME="docker-compose-example.tgz"
mkdir example && cd example
wget --no-check-certificate "https://docs.google.com/uc?export=download&id=${FILE_ID}" -O
${FILENAME}
tar xzf ${FILENAME}
cd ${CURRENT_WORKDIR}
```

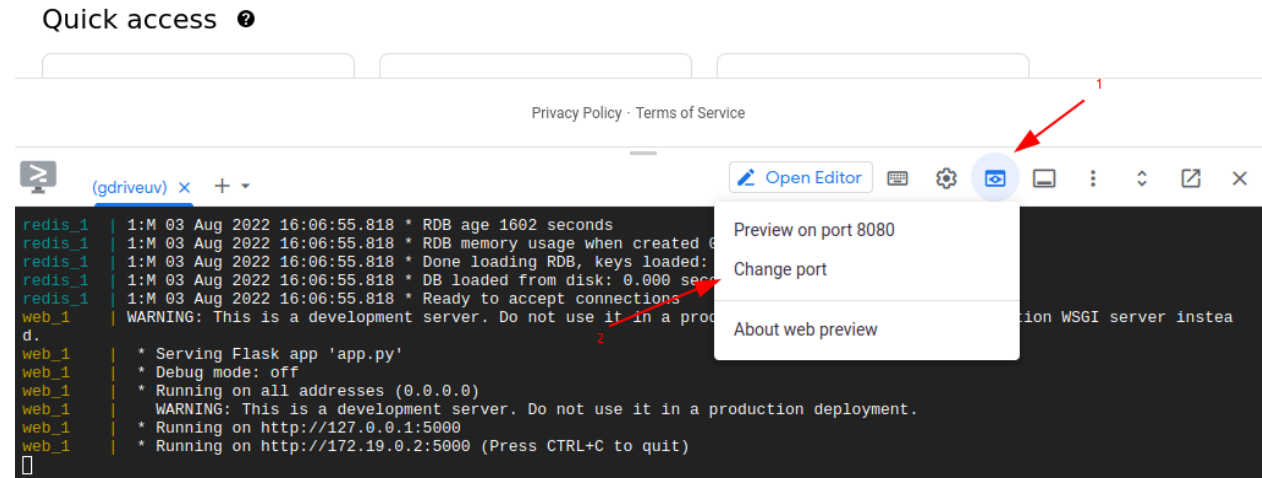
Cambie los permisos de ejecución del script y ejecútelo, lleve a cabo la ejecución de estos comandos:

```
chmod +x script.sh && ./script.sh
```

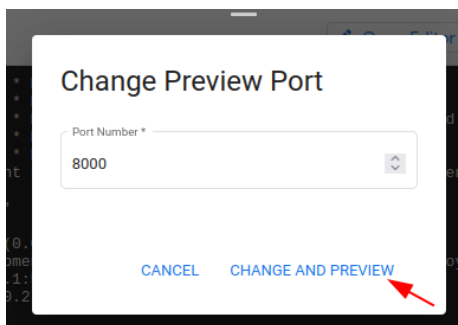
Este script crea un directorio llamado ‘example’. Ingrese al directorio ‘example’ y ejecute el comando ‘docker-compose up’ de la siguiente manera:

```
cd example && docker-compose up
```

Una vez se haya detenido la ejecución del comando 'docker-compose up', haga lo siguiente:



1. De 'click' sobre el ícono que luce como un navegador.
2. De 'click' luego sobre 'Change port' y escriba '8000'



Finalmente de 'click' sobre 'CHANGE AND PREVIEW'. Se abrirá una nueva pestaña y verá un mensaje parecido a este

Hello World! I have been seen 4 times.

## Servidor web de Apache

En la tercera máquina virtual (192.168.28.32) usted deberá instalar el servidor web de Apache. En Internet hay muchas páginas que muestran como llevar a cabo la instalación del servidor web de Apache.

# Algunos comentarios

## Respecto a la ejecución del ejercicio

Sugiero que se comience por llevar a cabo aquellas tareas más sencillas:

- Instalación del servidor web de Apache
- Despliegue y ejecución de aplicación basado en 'docker-compose'
- Instalación y configuración de Nginx como 'proxy inverso'

Estas tareas fueron listadas de menor a mayor grado de dificultad. Trate de llevar a cabo las tareas más sencillas de modo que gane experiencia y confianza en el proceso del desarrollo de este ejercicio.

## Respecto a la conectividad de las máquinas

Un punto importante es que, para el ejercicio, las máquinas virtuales se conectarán a través de una red virtual provista por la herramienta de virtualización. En VirtualBox el tipo de red que usarán las máquinas virtuales es la conocida como [host-only adapter](#). En KVM se puede crear una interfaz de red tipo bridge y a través de esa interfaz de red es que se logra la conexión entre las diferentes máquinas virtuales que se ejecutan bajo un mismo equipo anfitrión<sup>2</sup>.

## Entregables

El informe del ejercicio se debe entregar en documento en PDF, **No se aceptarán archivos de otro tipo.**

El documento deberá indicar claramente el nombre de los integrantes y presentar una tabla de contenido. El documento deberá presentar un informe en el cual se describa como se llevaron y alcanzaron cada uno de los objetivos planteados en el presente documento. Así mismo es muy importante documentar los problemas, desafíos que se presentaron durante el desarrollo del ejercicio y mostrar cómo estos desafíos se superaron. Indicar las fuentes de información en las que se apoyaron para llevar a cabo el desarrollo del ejercicio es parte importante del informe.

El documento deberá proveer **enlaces a los recursos** que sustentan el logro de los objetivos planteados por la tarea:

- Presentación en clase o video de Youtube, no superior a 20 minutos, y que sirva como una demostración de que el ejercicio funciona. Es importante que se evidencie también la participación de todos los miembros del grupo en el video. Pueden dividirse la presentación del video en segmentos donde cada miembro del grupo haga su participación. El video debe salir sin errores, es decir, que debe llevarse una presentación altamente depurada y ensayada.
- Repositorio en GitHub donde se muestre la documentación técnica que presenta los insumos que se usaron para lograr los objetivos planteados por la tarea. Aquí van los

---

<sup>2</sup> <https://superuser.com/questions/549350/host-only-like-interface-in-kvm>

códigos de la aplicación. También puede incluir algún tipo de script que le haya permitido automatizar una o varias partes del desarrollo del ejercicio.

## Número de integrantes

Los ejercicios se deben llevar a cabo en grupo. El número máximo de integrantes es de tres (3).

## Fecha de entrega

Según como se establezca en BrightSpace