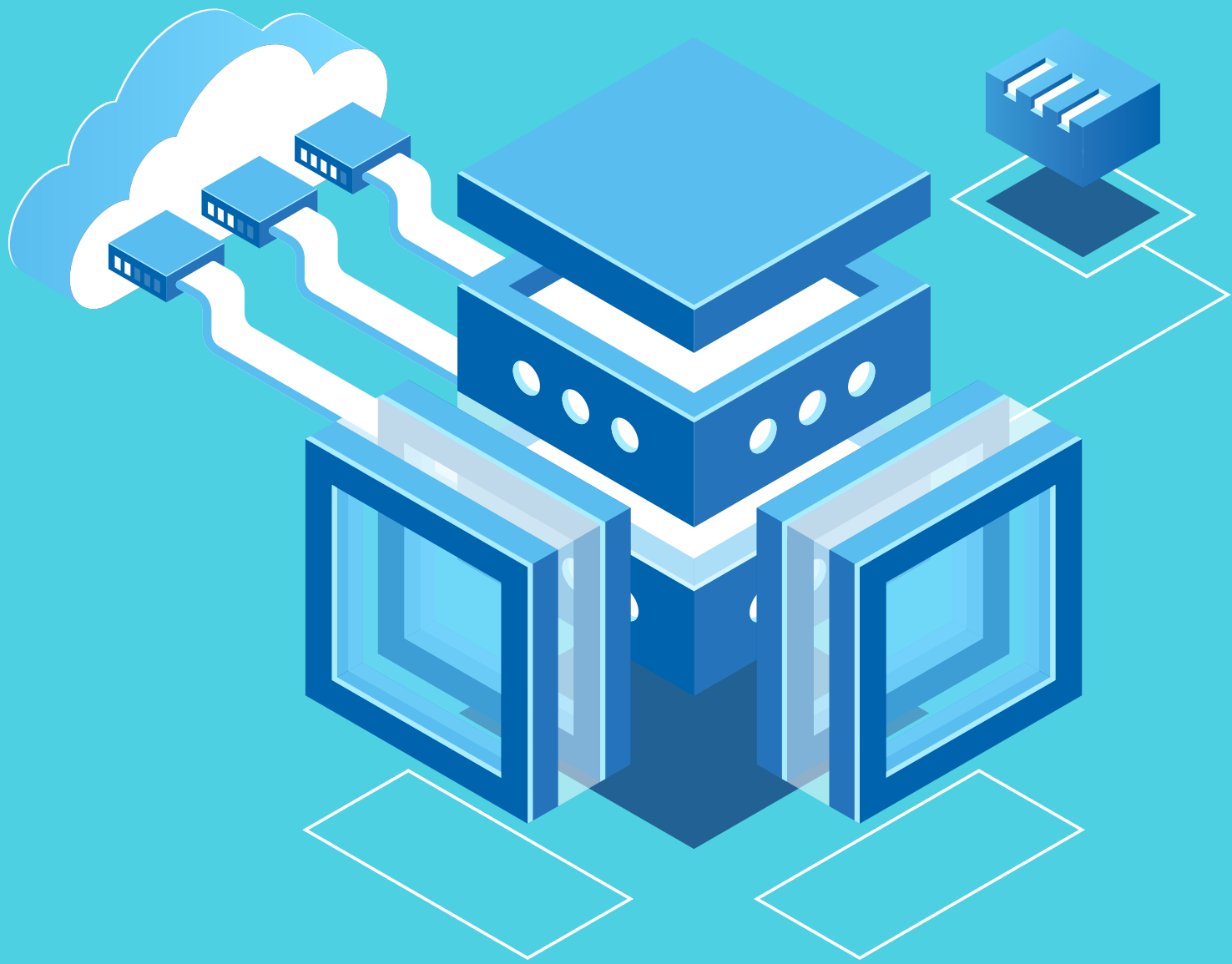


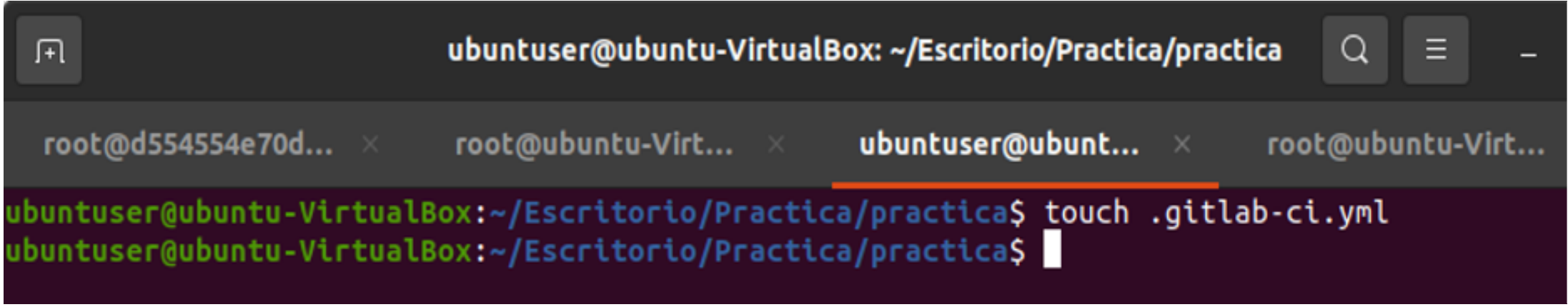
Repositorio central para GitLab



1 En primer lugar se debe crear el archivo **.gitlab-ci.yml** dentro del mismo directorio del ambiente de desarrollo para que sea enviado en un push al repositorio central.

Figura 1

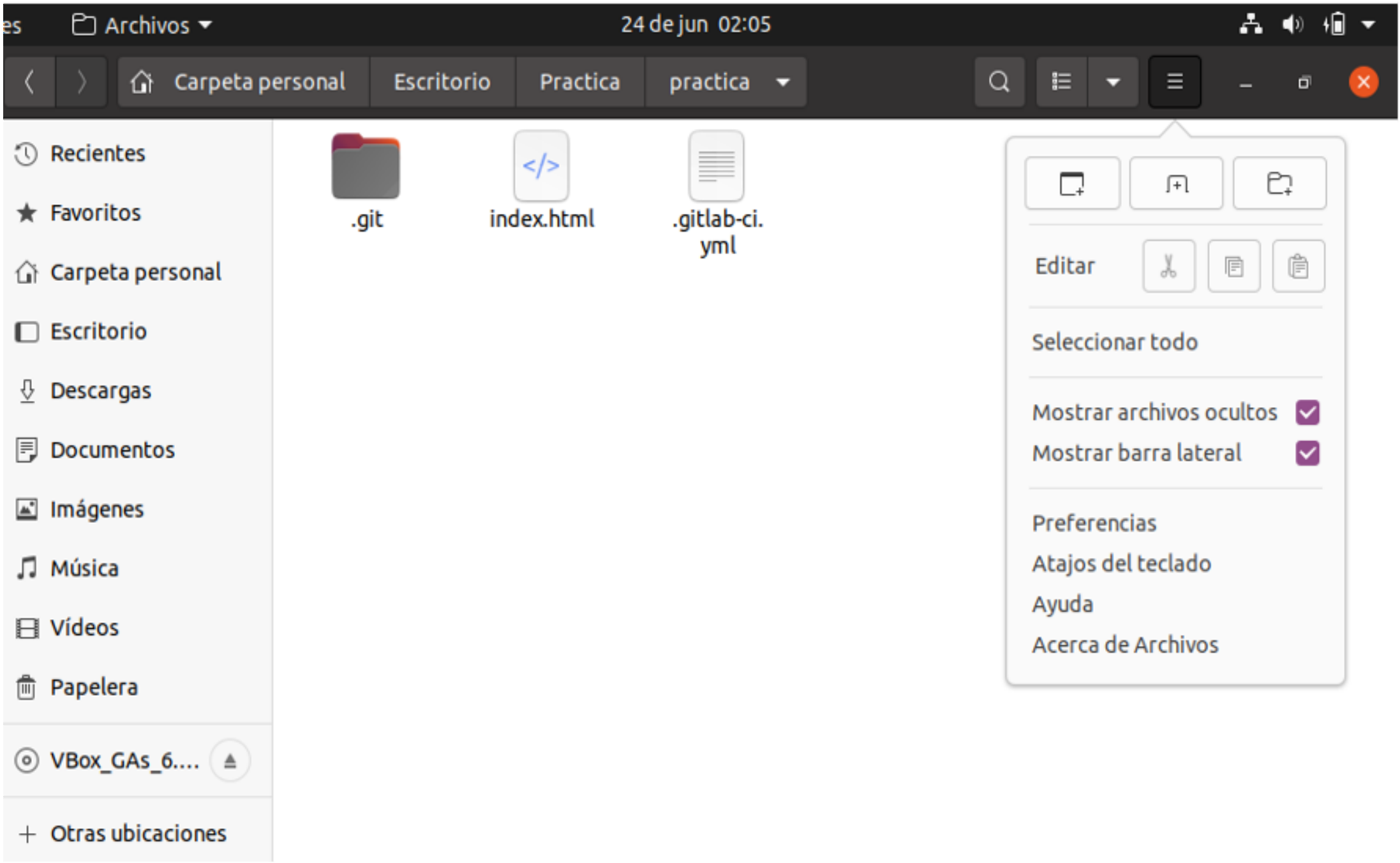
Creación de archivos



2 Como es un archivo oculto deberá asegurarse de visualizarlo para verificar su estructura y editarlo más fácilmente.

Figura 2

Verificación de estructura



3 Este archivo de extensión yml describirá el pipeline a ser ejecutado en GitLab para automatizar el proceso de entrega continua, pero puede describir cualquier actividad que desee ser automatizada. En esencia se definen estados, los cuales describen tareas.

Para este ejemplo se definen dos estados, uno inicial con una tarea de escribir un saludo y otro estado, con la tarea de replicar las acciones en el servidor web para actualizar el contenido de la carpeta desplegada de acuerdo con el repositorio central. (Labor que se realizó anteriormente de forma manual).

Figura 3

Acciones en el servidor web

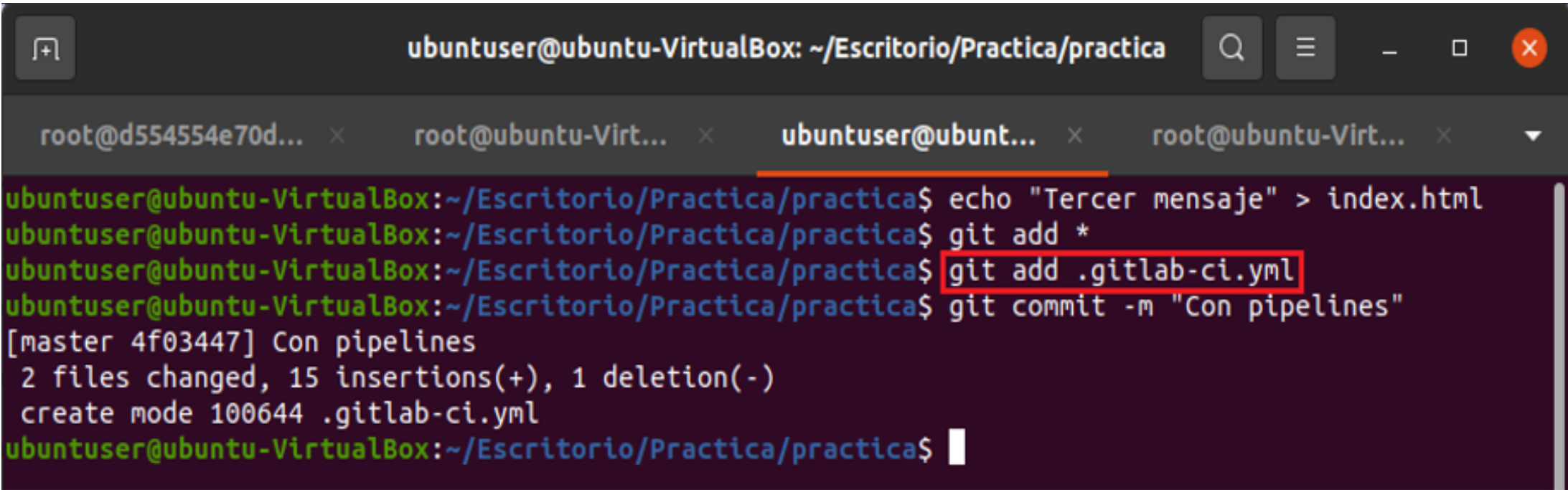


- 4
4. Note que el trabajo definido indica paso a paso lo que debe realizar en el servidor web. En este caso, lo que debe realizar primero es ingresar a la carpeta de despliegue por medio del comando `cd /var/www/html` y sobre esta carpeta ejecutar un proceso de pull para actualizarse de acuerdo con la información del repositorio central.

El archivo `.gitlab-ci.yml` debe ser enviado al repositorio central manualmente, ya que al ser un archivo oculto es omitido en las operaciones de preparación.

Figura 4

Actualización de información



- 5
- Es importante ahora instalar y ejecutar un servicio adicional en el servidor web que entienda las órdenes a ejecutar que serán enviadas desde el repositorio central, el cual se conoce como un **runner**.

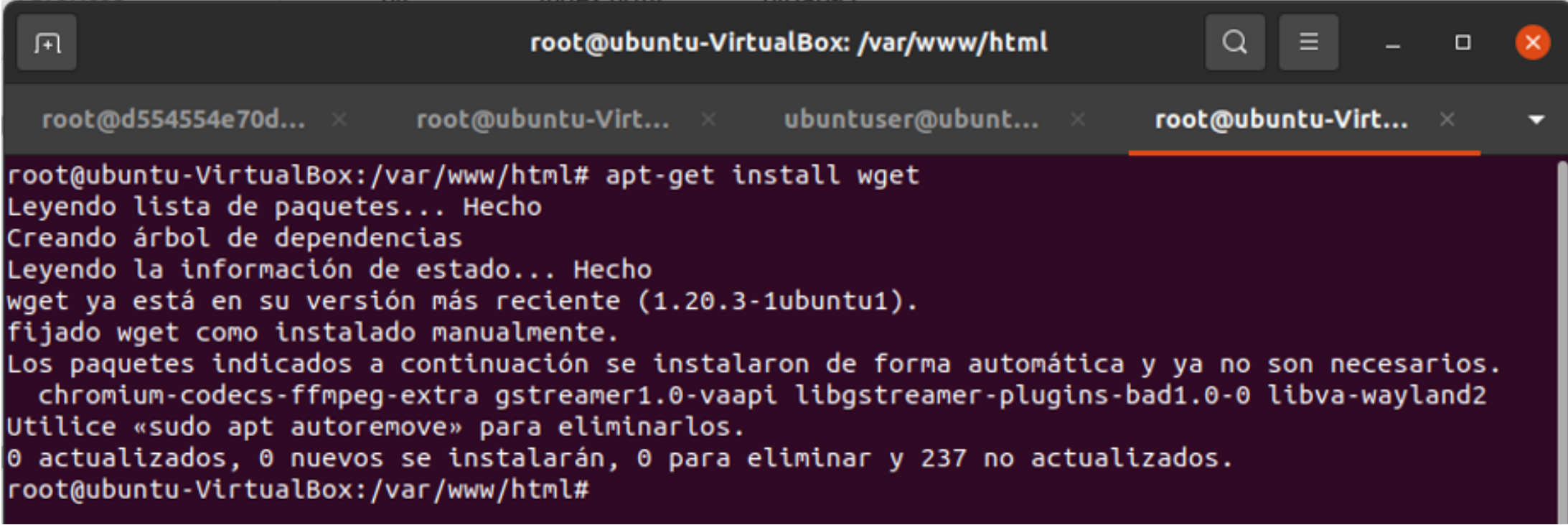
GitLab runner es un proyecto de código abierto que permite ejecutar trabajos y recibir información de los resultados de estas ejecuciones en **GitLab (Alex, 2019)**.

La instalación de este runner en la máquina del servidor web requiere la instalación de **wget**, si aún no lo tiene instalado lo puede hacer por medio del siguiente comando:

```
apt-get install wget
```

Figura 5

Instalación de wget

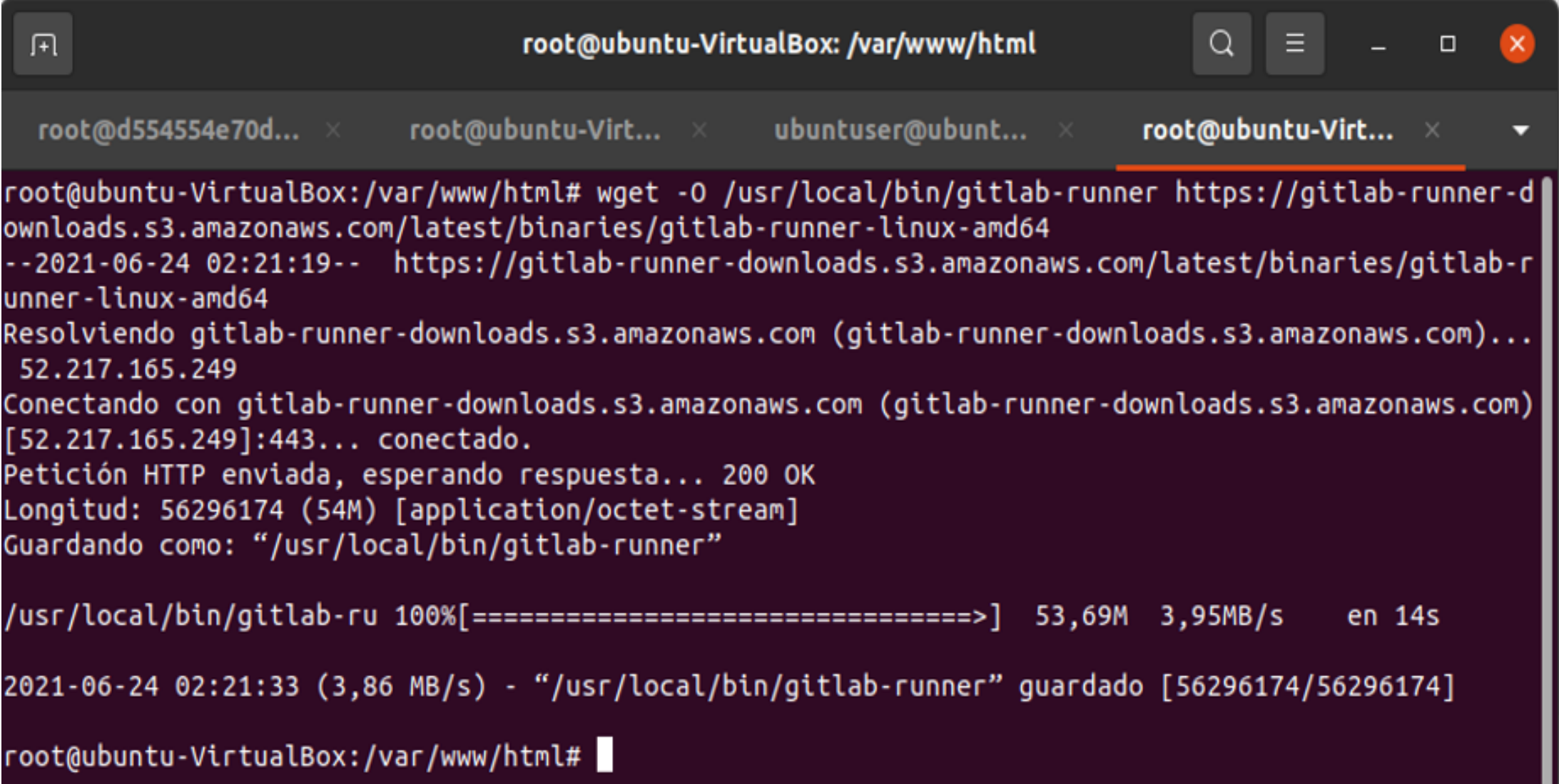


- 6
- Para instalar el runner de GitLab se ejecuta el siguiente comando:

```
wget -O /usr/local/bin/gitlab-runner https://gitlab-runner-downloads.s3.amazonaws.com/latest/binaries/gitlab-runner-linux-amd64
```

Figura 6

Instalación runner

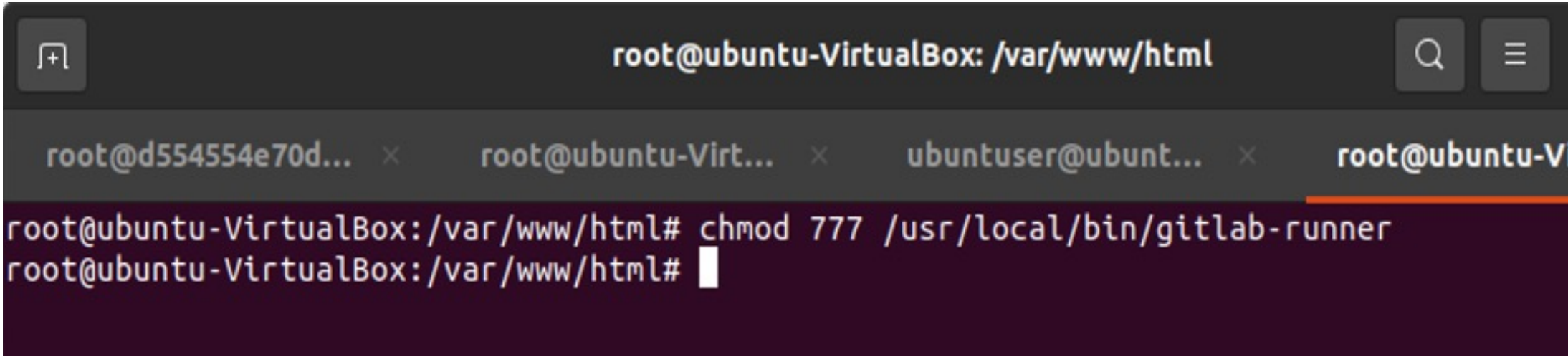


7 Luego se asignan los permisos requeridos con el comando:

```
chmod 777 /usr/local/bin/gitlab-runner
```

Figura 7

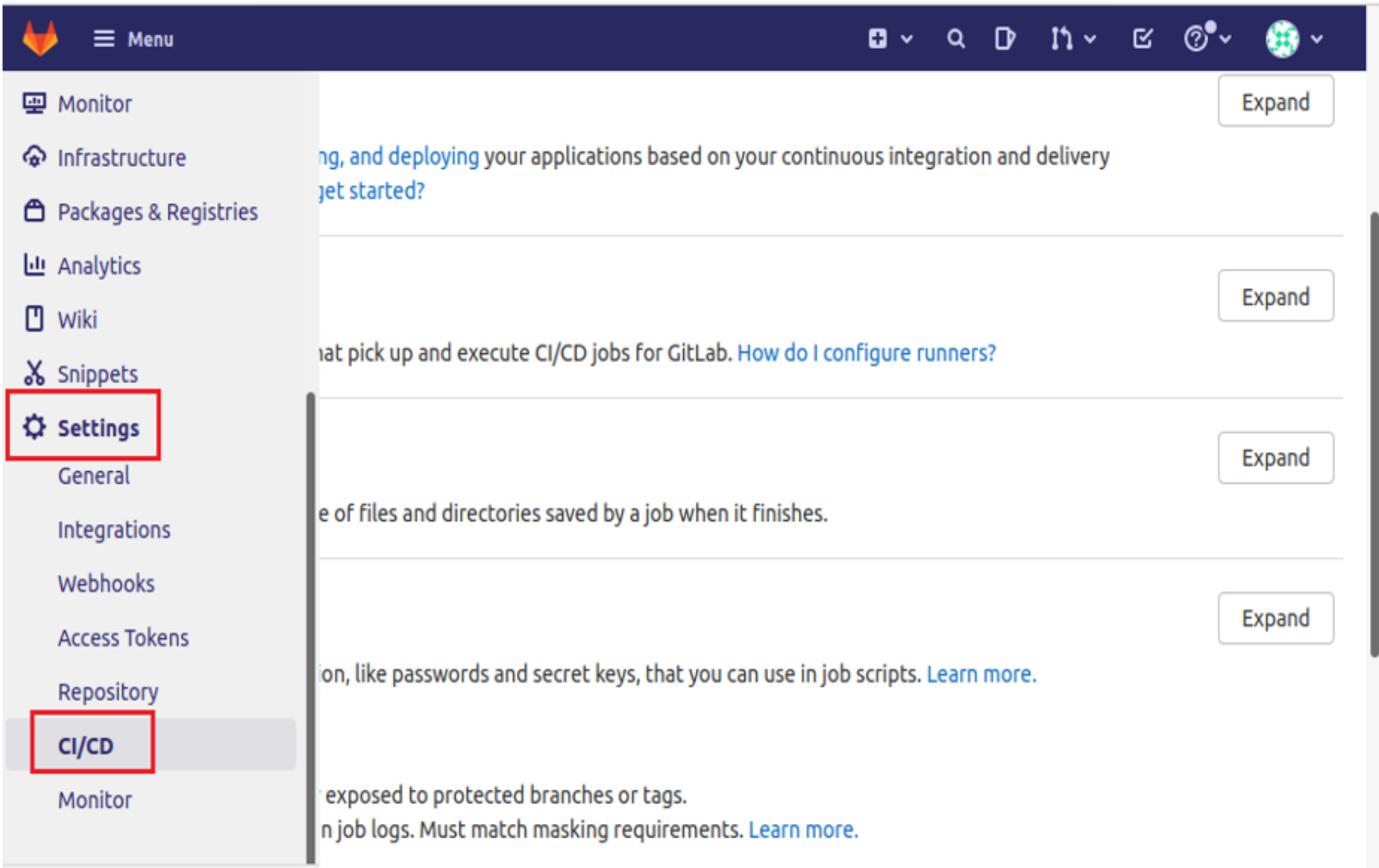
Permisos



8 También es necesario registrar el runner que corre en el servidor web con los datos donde se está ejecutando GitLab, que centraliza el repositorio y su respectivo token. Para esto, se debe ingresar en GitLab a configuración CI/CD.

Figura 8

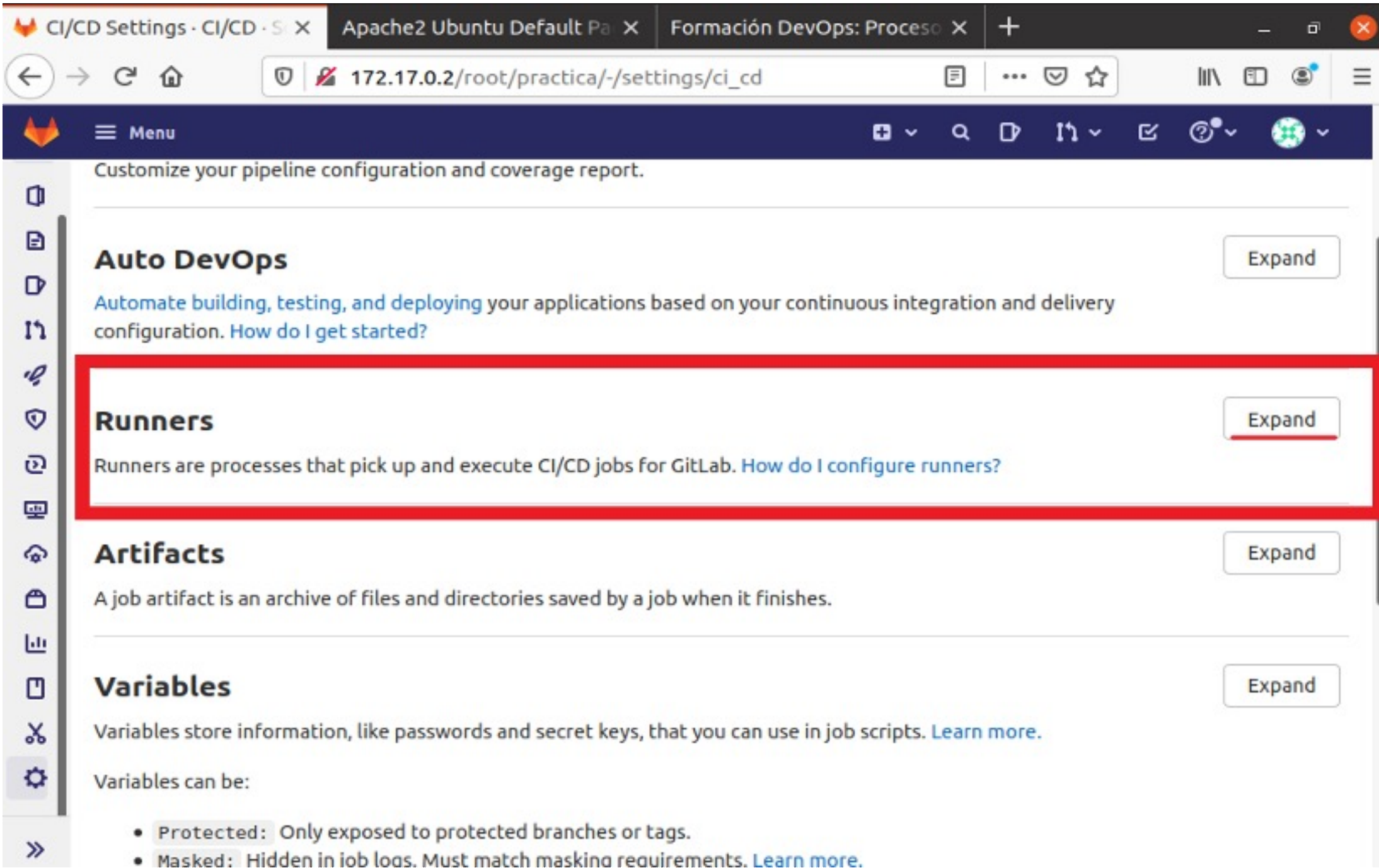
Ingreso a GitLab



9 Buscar y expandir el ítem runners.

Figura 9

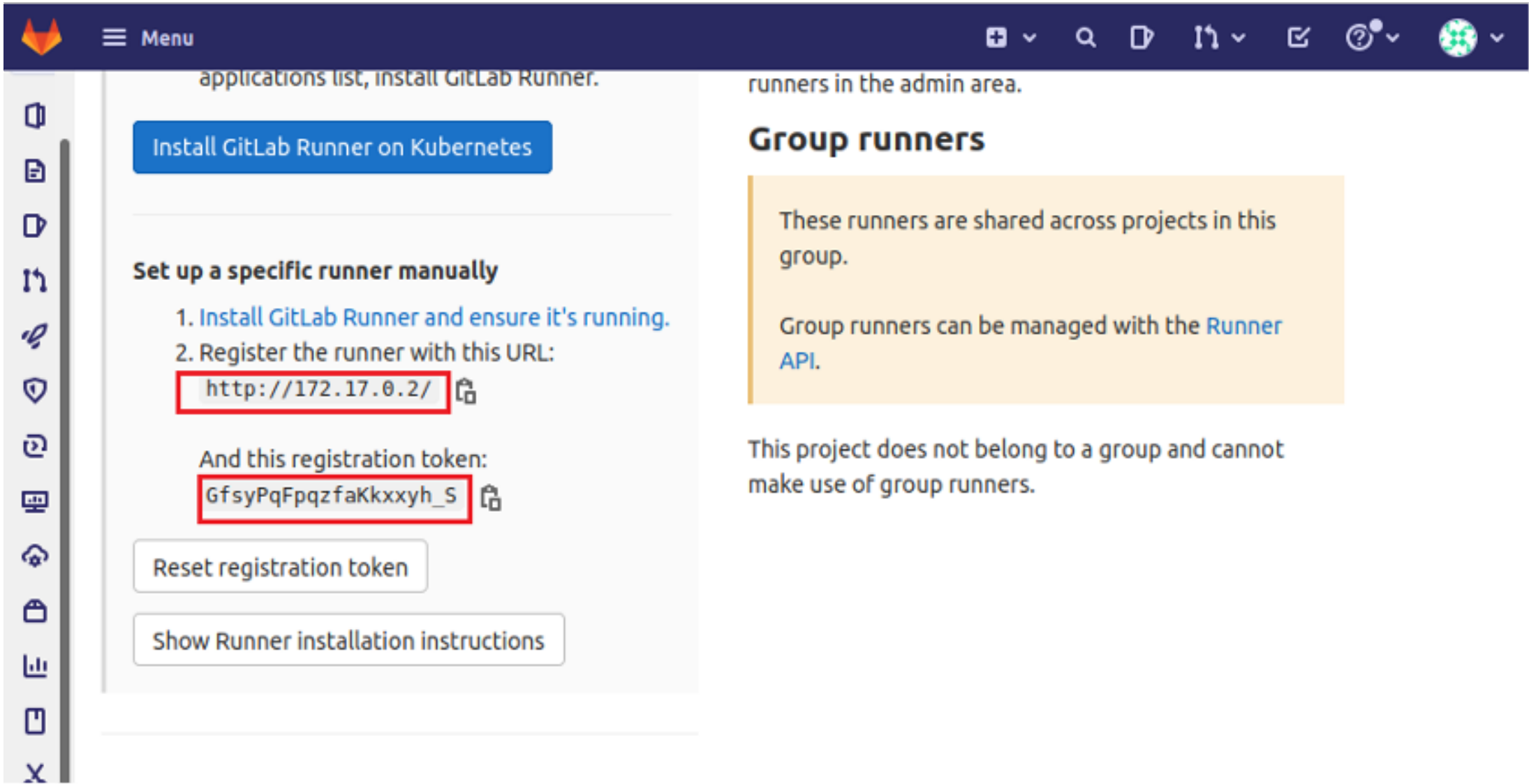
Expansión de runners



10 Sobre este ítem en la parte inferior se encuentran los datos necesarios para registrar el runner que se ejecutará en el servidor web.

Figura 10

Registro de runners



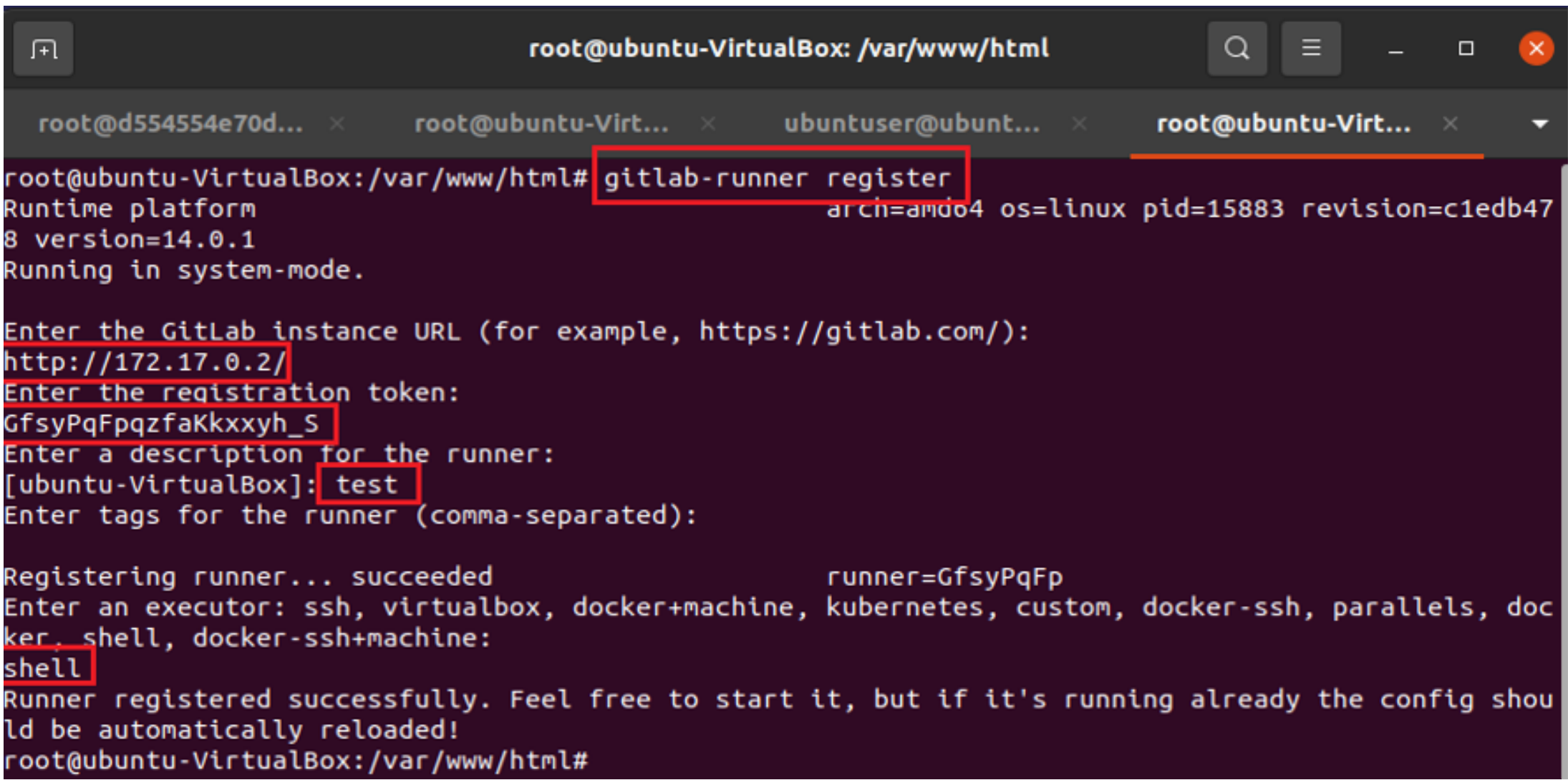
11 Teniendo esos datos disponibles se ejecuta el runner en el servidor web por medio del comando:

```
gitlab-runner register
```

12 Este comando solicitará la URL de la instancia de GitLab y el token, que fueron consultados anteriormente. Luego, un nombre con el cual se registrará y opcionalmente los tag a ser usados y finalmente, el modo de ejecución de este ejemplo que es shell.

Figura 11

URL de la instancia GitLab



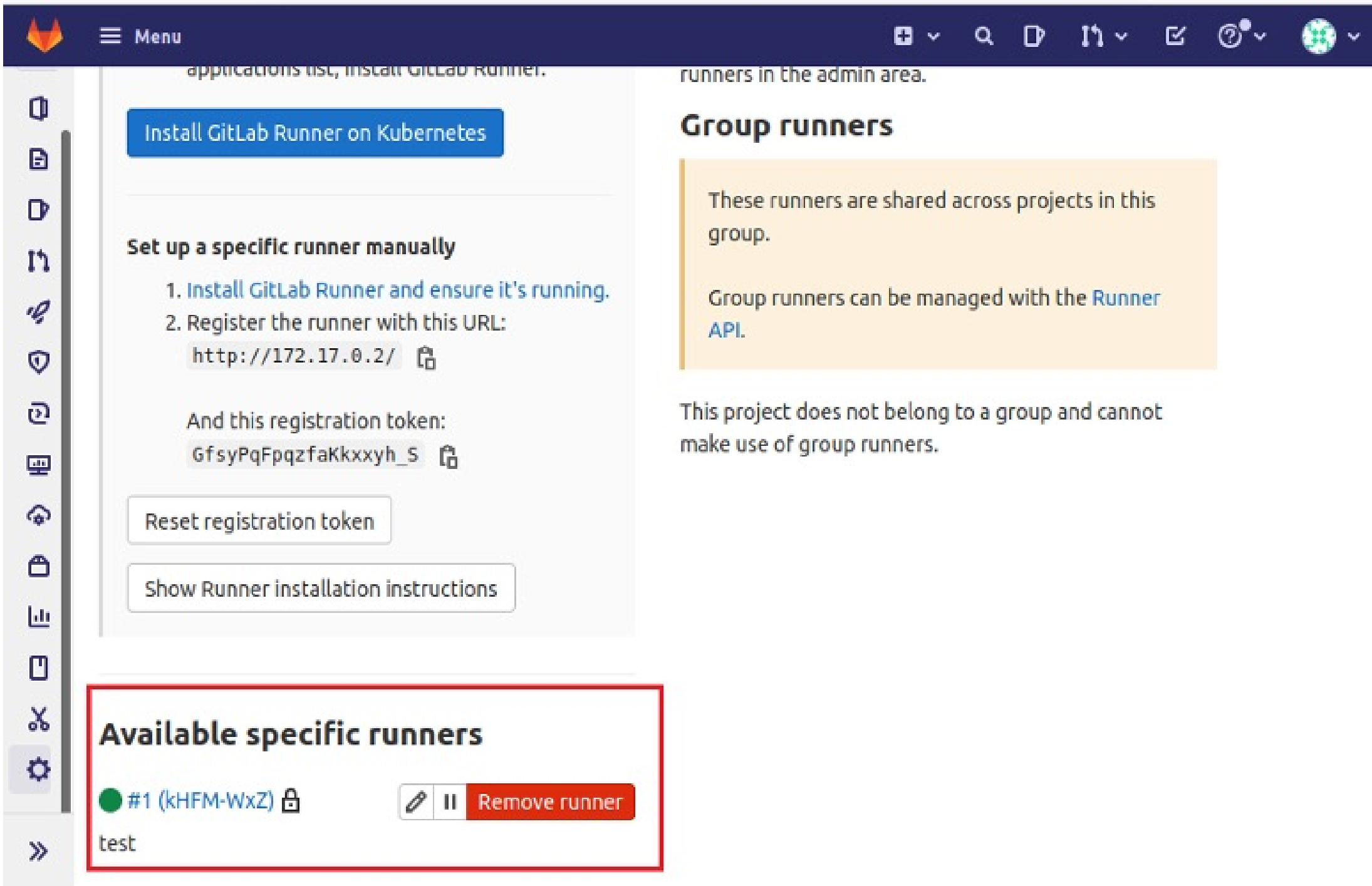
13 Luego se inicia el proceso con el comando:

```
gitlab-runner run
```

14 Una vez se ejecuta, se puede verificar en GitLab que la asociación fue satisfactoria.

Figura 12

Verificación de asociación



En este punto todos los cambios que sean confirmados en el repositorio central automáticamente serán desplegados en el servidor web, sin necesidad de que se ejecuten comandos sobre él.