



INSTITUTO POLITÉCNICO NACIONAL

Unidad Profesional Interdisciplinaria en Ingeniería y  
Tecnologías Avanzadas

## *Ejercicio de práctica 7*

Asignatura: Sistemas distribuidos

Profesor: Mata Rivera Miguel Félix

Alumnos:

- León Morales Valeria
- Romero Verdín Diether Sebastián
- Sánchez Morales Miriam

Grupo: 2TM9

Fecha: 29-abril-2022

# Práctica 7:

## INSTRUCCIONES:

Uso y aplicación del Framework (gRPC)

Entregar reporte con screenshots de los resultados

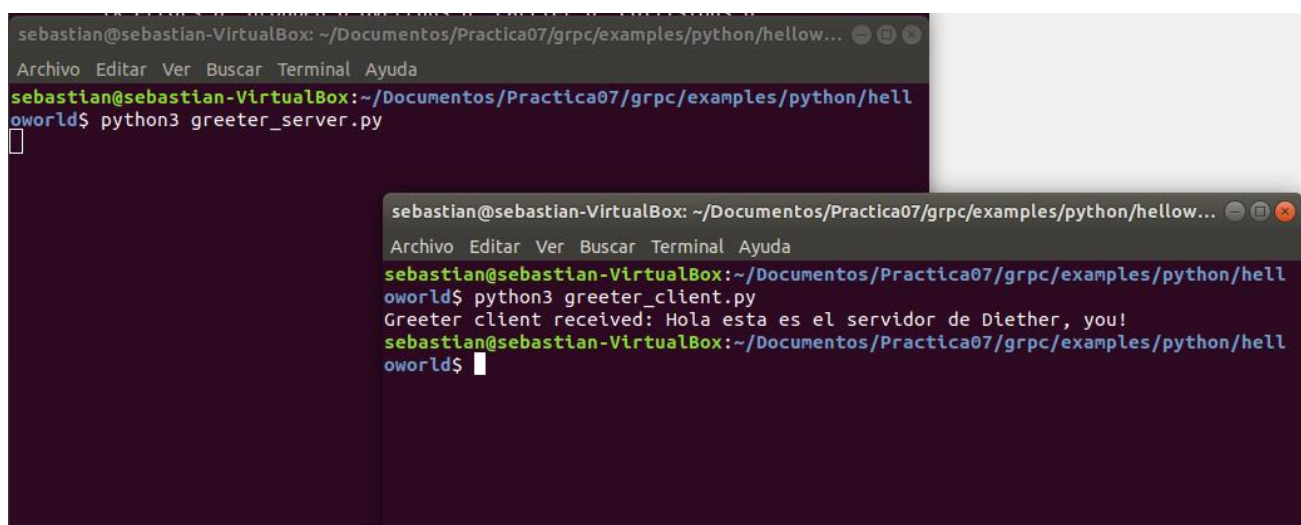
## Edición:

Para identificar al servidor que se conecta modificamos el mensaje del servidor en cada una de las máquinas:

```
class Greeter(helloworld_pb2_grpc.GreeterServicer):  
    def SayHello(self, request, context):  
        return helloworld_pb2.HelloReply(message='Hola esta es el servidor de Diether, %s!' % request.name)  
  
class Greeter(helloworld_pb2_grpc.GreeterServicer):  
    def SayHello(self, request, context):  
        return helloworld_pb2.HelloReply(message='Hola este es el servidor de mirimiri11, %s!' % request.name)
```

## Dos máquinas en la misma red

Hacemos una ejecución de manera local para verificar que funcione:

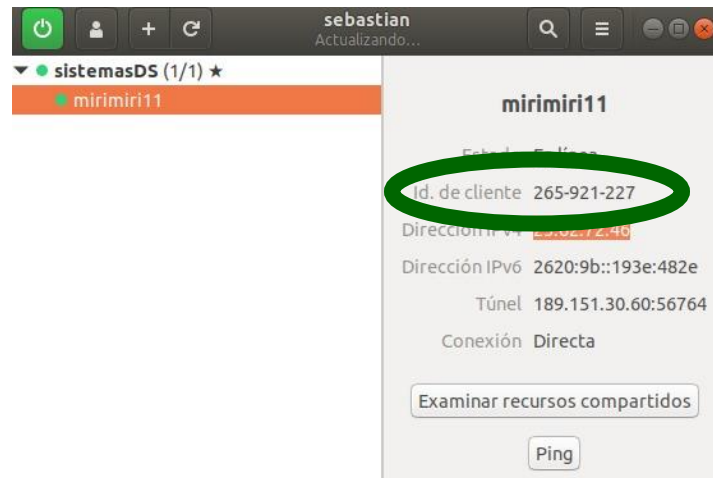


The image shows two terminal windows from a VirtualBox environment. The top window shows the execution of the greeter\_server.py script, which runs without any output. The bottom window shows the execution of the greeter\_client.py script, which outputs the message 'Greeter client received: Hola esta es el servidor de Diether, you!'.

```
sebastian@sebastian-VirtualBox: ~/Documentos/Practica07/grpc/examples/python/hellow...  
Archivo Editar Ver Buscar Terminal Ayuda  
sebastian@sebastian-VirtualBox:~/Documentos/Practica07/grpc/examples/python/hell  
oworld$ python3 greeter_server.py  
  
sebastian@sebastian-VirtualBox: ~/Documentos/Practica07/grpc/examples/python/hellow...  
Archivo Editar Ver Buscar Terminal Ayuda  
sebastian@sebastian-VirtualBox:~/Documentos/Practica07/grpc/examples/python/hell  
oworld$ python3 greeter_client.py  
Greeter client received: Hola esta es el servidor de Diether, you!  
sebastian@sebastian-VirtualBox:~/Documentos/Practica07/grpc/examples/python/hell  
oworld$
```

# Dos máquinas en diferente red

Ahora utilizamos haguichi para hacer la conexión entre dos máquinas de diferentes redes utilizando la ip que nos arroja:



Modificamos el código del cliente y colocamos la ip que arroja el programa en las dos máquinas:

```
def run():
    # NOTE(gRPC Python Team): .close() is possible on a channel and should be
    # used in circumstances in which the with statement does not fit the needs
    # of the code.
    #with grpc.insecure_channel('localhost:50051') as channel:
    with grpc.insecure_channel('25.62.72.46:50051') as channel:
        stub = helloworld_pb2_grpc.GreeterStub(channel)
        response = stub.SayHello(helloworld_pb2.HelloRequest(name='you'))
        print("Greeter client received: " + response.message)
```

```
def run():
    # NOTE(gRPC Python Team): .close() is possible on a channel and should be
    # used in circumstances in which the with statement does not fit the needs
    # of the code.
    #with grpc.insecure_channel('localhost:50051') as channel:
    with grpc.insecure_channel('25.1.126.62:50051') as channel:
        stub = helloworld_pb2_grpc.GreeterStub(channel)
        response = stub.SayHello(helloworld_pb2.HelloRequest(name='you'))
        print("Greeter client received: " + response.message)
```

# Funcionamiento:

## Servidor:

```
mirimiri11@mirimiri11-VirtualBox: ~/Documentos/P7/grpc/examples/python/helloworld
Archivo Editar Ver Buscar Terminal Ayuda
mirimiri11@mirimiri11-VirtualBox:~/Documentos/P7/grpc/examples/python/helloworld
$ python3 greeter_server.py
```

## Cliente:

```
sebastian@sebastian-VirtualBox: ~/Documentos/Practica07/grpc/examples/python/hellow...
Archivo Editar Ver Buscar Terminal Ayuda
sebastian@sebastian-VirtualBox:~/Documentos/Practica07/grpc/examples/python/hell
oworld$ python3 greeter_client.py
Greeter client received: Hola este es el servidor de mirimiri11, you!
sebastian@sebastian-VirtualBox:~/Documentos/Practica07/grpc/examples/python/hell
oworld$
```

Ahora hacemos el proceso inverso, ejecutamos el servidor en la maquina sebastian y el cliente en la maquina mirimiri11:

## Servidor:

```
sebastian@sebastian-VirtualBox: ~/Documentos/Practica07/grpc/examples/python/hellow...
Archivo Editar Ver Buscar Terminal Ayuda
sebastian@sebastian-VirtualBox:~/Documentos/Practica07/grpc/examples/python/hell
oworld$ python3 greeter_server.py
```

## Cliente:

```
mirimiri11@mirimiri11-VirtualBox: ~/Documentos/P7/grpc/examples/python/helloworld
Archivo Editar Ver Buscar Terminal Ayuda
mirimiri11@mirimiri11-VirtualBox:~/Documentos/P7/grpc/examples/python/helloworld
$ python3 greeter_client.py
Greeter client received: Hola esta es el servidor de Diether, you!
mirimiri11@mirimiri11-VirtualBox:~/Documentos/P7/grpc/examples/python/helloworld
$
```

Básicamente, en esta ocasión se realizó una práctica utilizando gRPC para Python. ¡El cliente llama al servidor para recibir un mensaje el cual es “Hello, you!”. Como se muestra anteriormente se realizó la actividad satisfactoriamente. Esto lo podemos comprobar con las capturas de pantalla anexadas que prueban la funcionalidad de la práctica, esto es apenas el primer paso para comenzar a utilizar las herramientas que el gRPC nos brinda para optimizar un sistema distribuido.