Distributed System Practical Work 2

RPC File Transfer

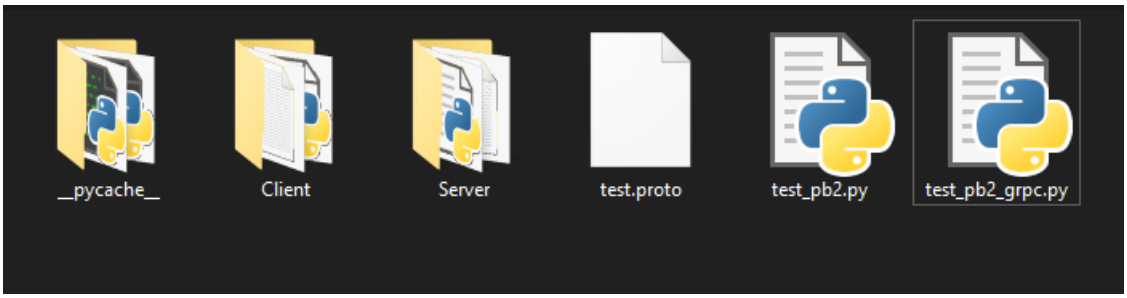
March 30, 2020

Contents

1 Protocol Design

This RPC Files Transfer system is currently only 1 way, we haven't found a way to implement 2 ways transfer, but it's a first step. In this system, only the Client send Files and the Server receives them. The Client can send however many files he wants.

We use gRPC protocol by Google. The first step is to produce a .proto file and then use the protoc compiler to get the "stub". Then using files produced by the proto compiler, we implement the gRPC files transfer, this is how our directory looks:



2 Codes

2.1 Server

```python
import sys
sys.path.append("F:\Atom Projects")

import test_pb2_grpc
import test_pb2
import grpc
from concurrent import futures
import numpy as np

f = open("server.txt", "r")
data = f.read()
f.close()

class Greeting(test_pb2_grpc.GreetingServicer):

    def SayHello(self, request, context):
        return test_pb2.HelloReply(message='Hello, %s!' % request.name)

    def SayHelloAgain(self, request, context):
        return test_pb2.HelloReply(message='Hello again, %s!' % request.name)

    def SendFile(self, request, context):
        filename = request.fileName
        f = open(filename, "w")
        f.write(request.fileContent)
        f.close()
        return test_pb2.HelloReply(message="The server have received your file: %s" % request.fileNa


server = grpc.server(futures.ThreadPoolExecutor(max_workers=10))
test_pb2_grpc.add_GreetingServicer_to_server(Greeting(), server)

print('Starting server. Listening on port 50051.')
server.add_insecure_port('[::]:50051')
server.start()
server.wait_for_termination()
```

2.2 Client

```python
import sys
sys.path.append("F:\Atom Projects")

import grpc
import test_pb2
import test_pb2_grpc

filename = input("Please enter the File's Name: ")
f = open(filename, "r")
content = f.read()
f.close()

with grpc.insecure_channel('localhost:50051') as channel:
    stub = test_pb2_grpc.GreetingStub(channel)
    response = stub.SayHello(test_pb2.HelloRequest(name='you'))
    response3 = stub.SendFile(test_pb2.File(fileName=filename, fileContent=content))

print("Greeter client received: " + response.message)
print("Greeter client received: " + response3.message)
```

2.3 Result on CMD

Client side

```
(nig) F:\Atom Projects\Client>python rpc_client.py
Please enter the File's Name: client.txt
Greeter client received: Hello, you!
Greeter client received: The server have received your file: client.txt
```

Server side

```
(nig) F:\Atom Projects\Server>python rpc_server.py
Starting server. Listening on port 50051.
```

In conclusion:
As discussed above, this is only a 1 way files transfer, maybe in the future we will come back to this and implement a 2 ways files transfer.