Oregon State University
Ecampus

Brian Dy

CS361: Assignment 5: Microservices Case Study & Pipe Spike

Overview

Learn how microservices work in the real world by (1) researching a software product that uses the microservices architecture and (2) implementing a microservices communication pipe that is NOT text files.

Instructions

Complete each item below by replacing the highlighted text (**Usability note**: double-click the text to select it).

1. **PART 1: Microservices Case Study**

   Find **well-known software** that uses the **microservices** architecture (e.g., Netflix, Amazon, etc.). **Research** the software and answer the following **questions**.

   a. What is the **name of the software** and **what is it for**?

   Amazon. Company is known for their seller marketplace, Prime video, and their AWS which provides cloud services for many companies such as Netflix, Spotify, Twitter (X).

   b. **Why** was the microservices architecture used for this software?

   *Amazon follows the microservice features vs monolithic architecture. As on their website for why microservices are utilized (*What are Microservices? | AWS (amazon.com)*), is that when using a monolithic approach, many of the features that are typically used become too complex as the code base continues to grows. As a result, the application becomes more difficult to implement and many of the features become too dependent on one another which can increase the risk of a single process failure.*
   However, when using microservices, applications are independent of one another and they each run their own application process as a service. In order for these independent applications to communicate, they utilize light weight API's that interface with each other. One of the big benefits of independently run applications is that they can each be updated, deployed, and/or scaled back to meet specific demands as needed without significant failures to the whole system.

   c. How does **communication** happen between the software's microservices?

   *Amazon has various mechanisms for how each of the microservices communicate with each other. They include using REST-based communication, GraphQL, gRPC, Asynchronous messaging and event passing, and Orchestration and state management (*Communication mechanisms - Implementing Microservices on AWS (amazon.com)*)*

REST-based communication – uses the HTTP/S protocol through RESTful APIs as well as AWS API gateway
GraphQL – uses synchronous communication similar to REST but limits the exposure to a single end point.
gRPC – uses HTTP/2 and has features such as compression and stream prioritization
Asynchronous messaging and event passing – sends and receives message via a queing system in order to promote service discovery
Orchestration and state management – uses a centralized approach known as "orchestrator" and is responsible for managing and coordinating the interactions between the various microservices.

d. Name and **describe a few microservices** that are part of the software. (3+ microservices)

Amazon deploys a couple of various microservices that can be divided into various components that include:

- *Compute*
  - *Containers – scalable microservice container management system*
  - *Serverless – known as AWS lambda that allows you to run code without managing a server. Just upload the code and lambda manages the server*
- *Storage & Databases (not all are included)*
  - *Elasticache – allows a user to retrieve data and mange their information from their in-memory cache system*
  - *Aurora – a relational DB system*
- *Logging and Monitoring*
  - *CloudTrail – API monitoring system that continuously allows you to monitor and retain account activity*
  - *AWS X-Ray – uses end-to-end view of requests as they travel through the application*

2. **PART 2: Pipe Spike**

Spike one microservices communication approach that is NOT communication via text file (since you already tried that). Example approach:

- RabbitMQ
- Subprocess calls
- Sockets
- PyZMQ Messaging
- RPyC: Remote Python Call library
- HTTP Request
- os library: system calls
- ZeroMQ
- FTP

You are NOT limited to the list above.

Requirements for the approaches:

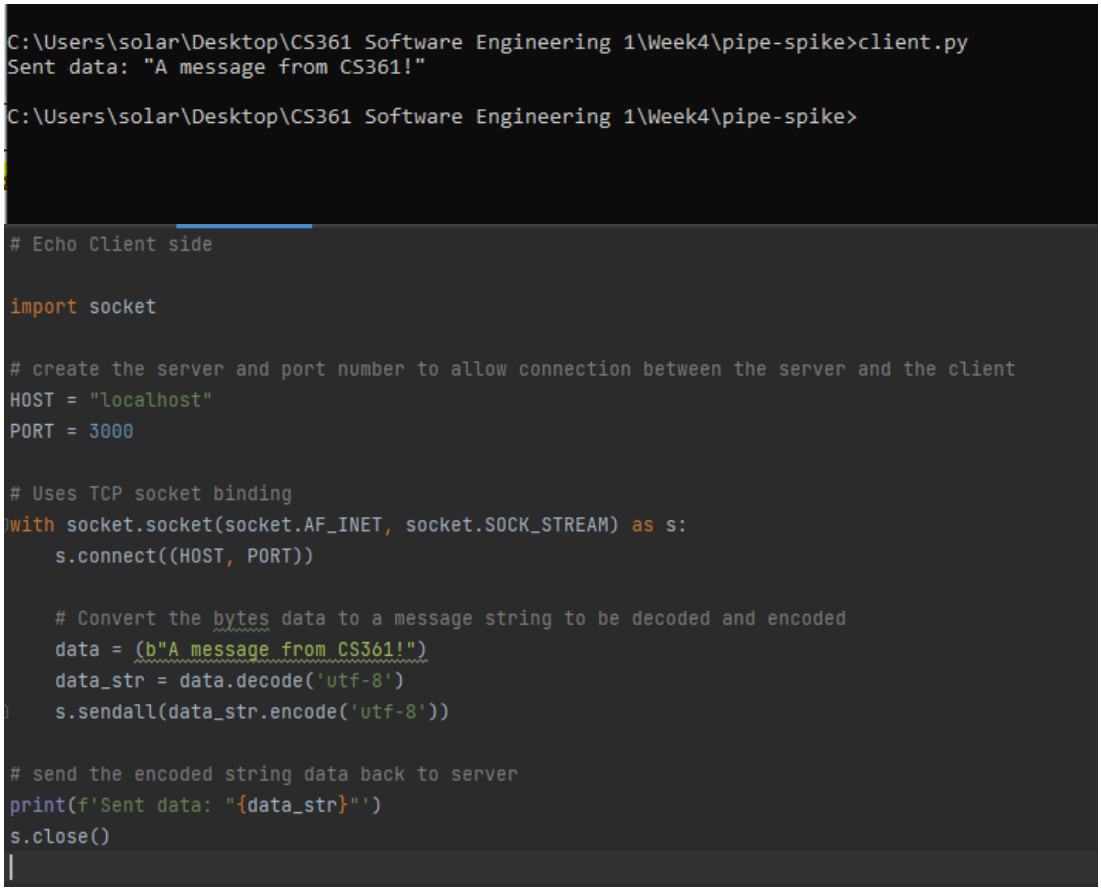
- Can be used to communicate between processes
- Can be used to request and provide data
- Not text files, CSV files, or other similar approaches involving file reads/writes

Complete the following:

a. **Which approach** did you spike?

Python Sockets using TCP protocol

b. Get the approach working. Upload **screenshots** that show the approach being used to **send and receive this message: ``A message from CS361''**



```
C:\Users\solar\Desktop\CS361 Software Engineering 1\Week4\pipe-spike>server.py
Server listening on ('localhost', 3000)
Waiting for connection

('127.0.0.1', 55541) established
Received:  A message from CS361!
Data has been received.
Closing connection at ('127.0.0.1', 55541)

C:\Users\solar\Desktop\CS361 Software Engineering 1\Week4\pipe-spike>
```

```python
# Echo Server Side

import socket

# create the server and port number to allow connection between the server and the client
address = ('localhost', 3000)

# create the TCP socket
server_socket = socket.socket(socket.AF_INET, socket.SOCK_STREAM)

server_socket.bind(address) # bind the address to the TCP socket
server_socket.listen(3)     # allows for u up to 3 connections to listen

print(f"Server listening on {address[0], address[1]}")

while True:
    print("Waiting for connection\n")
    conn, addr = server_socket.accept()

    print(f'{addr} established')
    while True:
        data = conn.recv(1024)
        if not data:
            break
        print("Received: ", data.decode('utf-8'))

    # close the socket once all data has been received
    print(f'Data has been received. \nClosing connection at {addr}')
    conn.close()
    break
```

Submission

PDF or Word format via Canvas.

You must follow instructions at Modules > “Attach a Document to "Text Entry" Field”.

Grading

You are responsible for satisfying all criteria listed in the Canvas rubric for this assignment. You will be able to revise this assignment if you miss points.

Questions?

Please ask via Ed so that others can benefit from the answers.