Homework 2 - Distributed Shared Memory

Course : Operating Systems
Deadline :Saturday , 2019/4/6 00:00

Overview

For inter process communication, there are two mechanisms can meet the need that includes message passing and shared memory approaches. In distributed computer environment, we can also use **distributed shared memory** and **message queue** to meet the need of cross-machine communication. In this assignment, we are going to implement a client-server communication service, which is based on a distributed shared memory architecture.

The Loot Box

Recent years, the loot box play a big role in the mobile game. In this project, we are going to implement the loot box server.

Tutorials

For this server, we are going to implement a distributed share memory server. This server should include:

1.Use multi-threading to implement distributed shared memory.

2.Prevent the race condition problem which means none of the clients will get any same drawing result (number).

3.A random list of the number 1 to 100,000 and use a counter to record the index of the list for current drawing result (number). Because the function of list, like pop(), is thread-safe, if you use those function, you will get no score in this homework.

4.Server can be connected by multiple client.

5.Server should send the drawing result (number) back to client.

Specification

We will provide sample “client.py”, you can use it to test your server with the following command.

python3 client.py [ip] [port]

If you want to export you output to a file for testing if race condition is happening. You can use the following command.

python3 client.py [ip] [port] > [file name]

Then send the following command to server then you will get the result of your draw.

draw [number]

Scenario:

Sever’s random list:[5,19,228,665,7,11,…]

Client A draw 4

Client B draw 2

1.With lock

Counter	0	1	2	3	4	5
Thread	A	A	A	B	A	B
Result	5	19	228	665	7	11

2.Without lock (Race condition!!!)

Counter	0	1	2	2	4	5
Thread	A	A	A	B	A	B
Result	5	19	228	228	665	7

File submission

Create your file with the name [student_id].xx (.py, .cpp, etc)

If you want to upload more than one file, zip it with the name [student].zip.

Upload it to the new E3 platform.

TA would validate your source codes by cheating detection. Please finish the assignment by yourself.

Note

- We have no limitation on the programming language.
- The loot box server should be executed on the AWS instance.

Reference

Echo server in Python

Echo server in C/C++