DISTRIBUTED SYSTEMS AND COMPUTTING ASSIGNMENT 1

Roll no:2023115125

Implement RPC and RMI in a Cloud Environment

1. Title

Implementation of RPC and RMI in a Cloud Environment using ngrok

---

2. Objective

The objective of this assignment is to implement a distributed application using **Remote Procedure Call (RPC)** and **Remote Method Invocation (RMI)**, where the server is hosted in a **cloud-accessible environment**. The client should be able to remotely access services provided by the server and receive responses successfully.

---

3. Technologies Used

- Programming Language:
  - RPC: **Python**
  - RMI: **Java**
- Cloud Exposure Tool: **ngrok**
- Operating System: Windows
- Networking Protocols: TCP/IP

---

4. System Architecture

The system follows a **client-server architecture**:

- The **server** hosts remote procedures and remote objects.
- The **client** invokes procedures/methods remotely.
- **ngrok** is used to expose the local server to the public cloud network.

---

5. RPC Implementation

5.1 Description

RPC allows a client to call functions on a remote server as if they were local. In this implementation:

- The server defines multiple remote procedures.
- The client sends requests to the server.
- The server processes the request and returns the result.

5.2 Steps Followed

1. Implemented RPC server with defined remote procedures.
2. Implemented RPC client to invoke server procedures.
3. Exposed the RPC server using **ngrok**.
4. Client connected to the ngrok public URL.
5. Server executed procedures and returned results.

5.3 Output

- Successful remote procedure invocation.
- Correct results received by the client from the cloud-hosted server.

---

6. RMI Implementation

6.1 Description

RMI enables Java programs to invoke methods on objects located on remote machines.

6.2 Steps Followed

1. Defined a **remote interface** extending `Remote`.
2. Implemented the interface in a remote class.
3. Registered the remote object using **RMI Registry**.
4. Exposed the RMI server using **ngrok**.
5. Client looked up the remote object and invoked methods.

6.3 Output

- Successful remote method invocation.
- Correct output returned from the server.

---

7. Cloud Hosting using ngrok

- ngrok was used to expose the local RPC and RMI servers to the internet.
- It provided a **public URL**, enabling cloud-based access without deploying on paid cloud services.
- No credentials were shared or uploaded.

---

8. Error Handling

- Network errors were handled using try-catch blocks.
- Invalid input and connection failures were managed gracefully.

---

9. Screenshots / Logs

- RPC client-server communication screenshots.
- RMI registry, server, and client execution screenshots.
- ngrok tunnel active screenshots.

---

10. Conclusion

This assignment successfully demonstrated the use of **RPC and RMI in a cloud environment**. Using ngrok simplified cloud exposure and allowed seamless remote communication between client and server applications.
