Remote Procedure Call (RPC) – Cloud-Based Calculator Application

1. Aim

To implement a Remote Procedure Call (RPC) based Calculator Application where the server is hosted on AWS EC2 and the client remotely invokes arithmetic procedures.

2. Objective

- To understand RPC-based communication in distributed systems
- To implement client–server interaction using RPC
- To deploy the RPC server on a cloud environment
- To verify remote procedure execution and result transfer

3. System Requirements

Hardware

- Computer with minimum 4 GB RAM
- Internet connection

Software

- Operating System: Ubuntu (Server), Windows / Kali Linux (Client)
- Programming Language: Python
- Cloud Platform: AWS EC2

4. RPC Architecture

RPC follows a procedure-oriented client–server architecture:

- Client invokes remote procedures like local function calls
- Server executes requested procedures

- Results are returned over the network
- Communication details are hidden from the programmer

---

5. RPC Implementation Details

5.1 Remote Procedures

The RPC server exposes arithmetic procedures that can be invoked remotely by the client. These procedures perform basic calculator operations:

- Addition – Adds two numbers
- Subtraction – Subtracts one number from another
- Multiplication – Multiplies two numbers
- Division – Divides two numbers with zero-division checking

Each procedure executes on the server and returns the computed result to the client.

5.2 Working Principle

The working of the RPC-based calculator application is as follows:

1. The RPC server defines arithmetic procedures such as addition, subtraction, multiplication, and division.
2. The server starts listening on a specific port for incoming client requests.
3. The client connects to the RPC server using the server's public IP address and port number.
4. The client invokes remote procedures as normal function calls.
5. The server processes the request, performs the computation, and sends the result back to the client.

This mechanism allows transparent execution of procedures across distributed systems.

---

6. Cloud Deployment

- RPC server is deployed on AWS EC2 Ubuntu instance
- Public IP address is used for client access
- Required ports are enabled in the EC2 security group

Screenshots to Attach:

✔ EC2 instance running
✔ Security group inbound rules

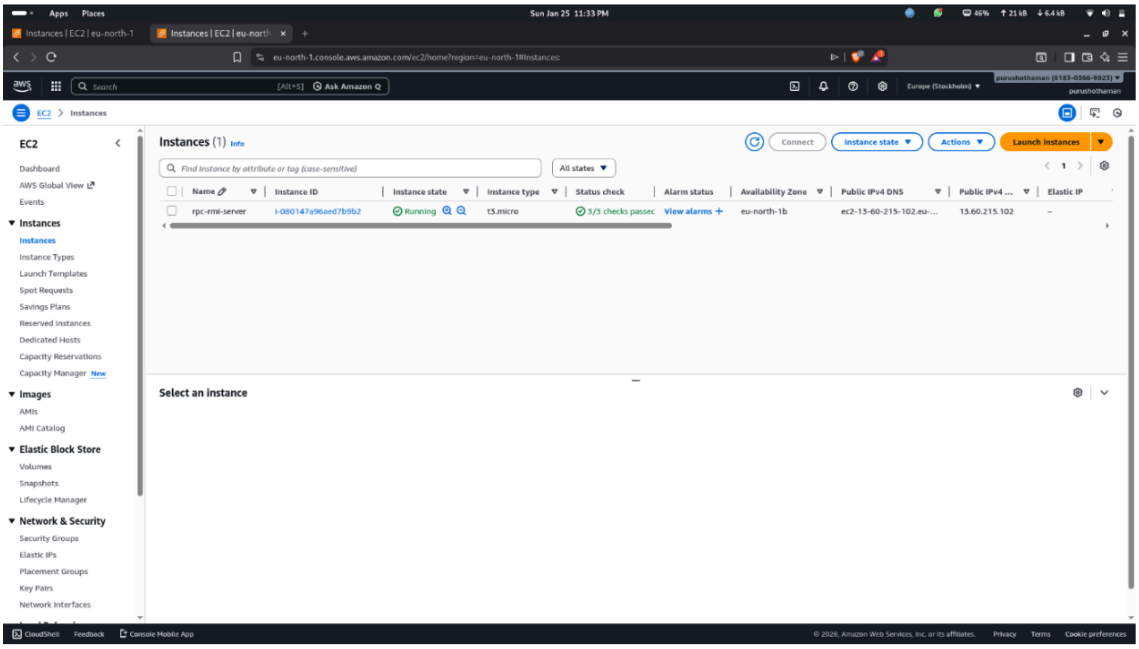
✔ RPC server execution
✔ Client output

7. Error Handling

- Invalid inputs are handled safely
- Division by zero is checked and prevented
- Network-related exceptions are managed using try–except blocks

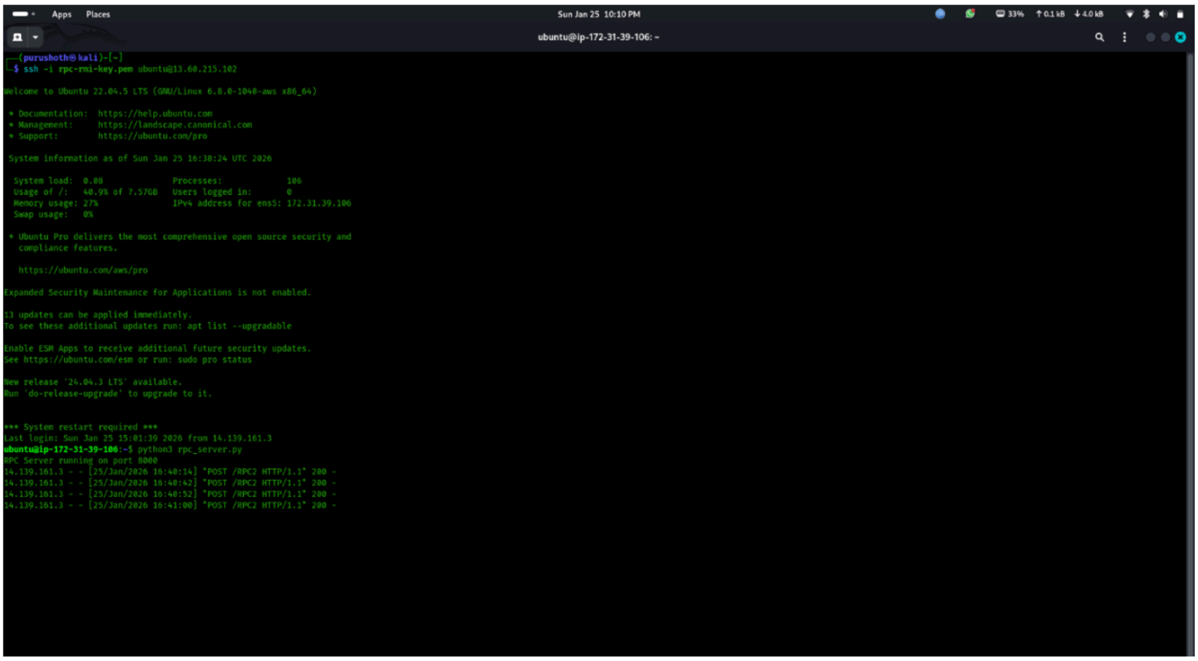
8. Output

- RPC server running on AWS EC2
- Client successfully invokes remote procedures
- Correct arithmetic results displayed



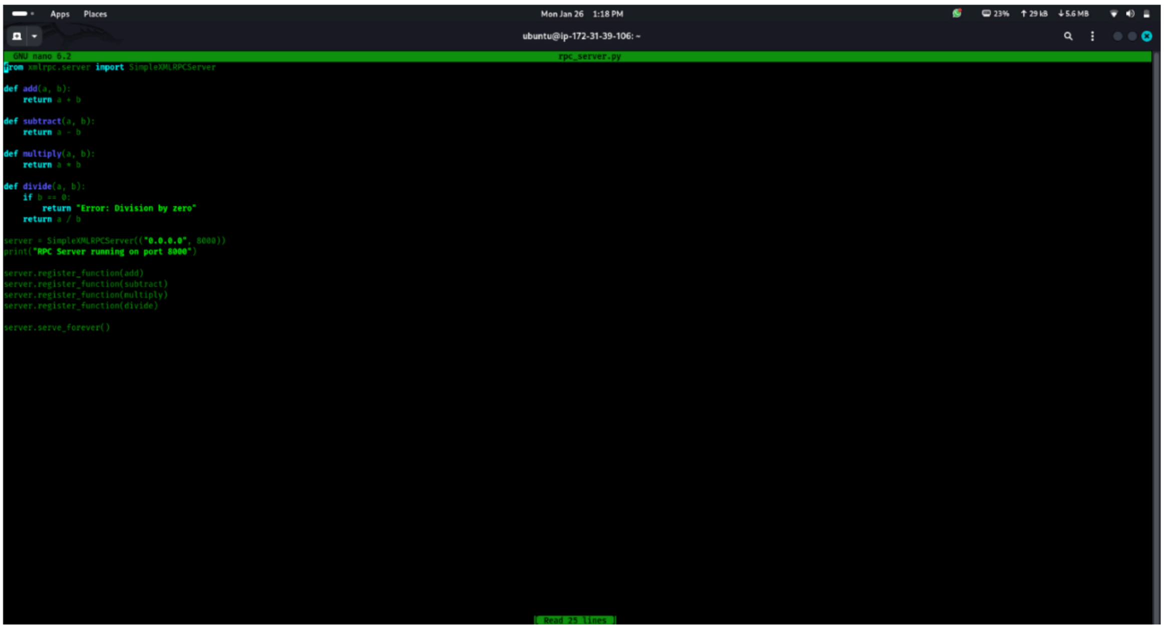
Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IPv4 ...	Elastic IP
rpc-rmi-server	i-080147a96aed7b9b2	Running	t3.micro	3/3 checks passec	View alarms +	eu-north-1b	ec2-13-60-215-102.eu-...	13.60.215.102	–

RPC server running on AWS EC2



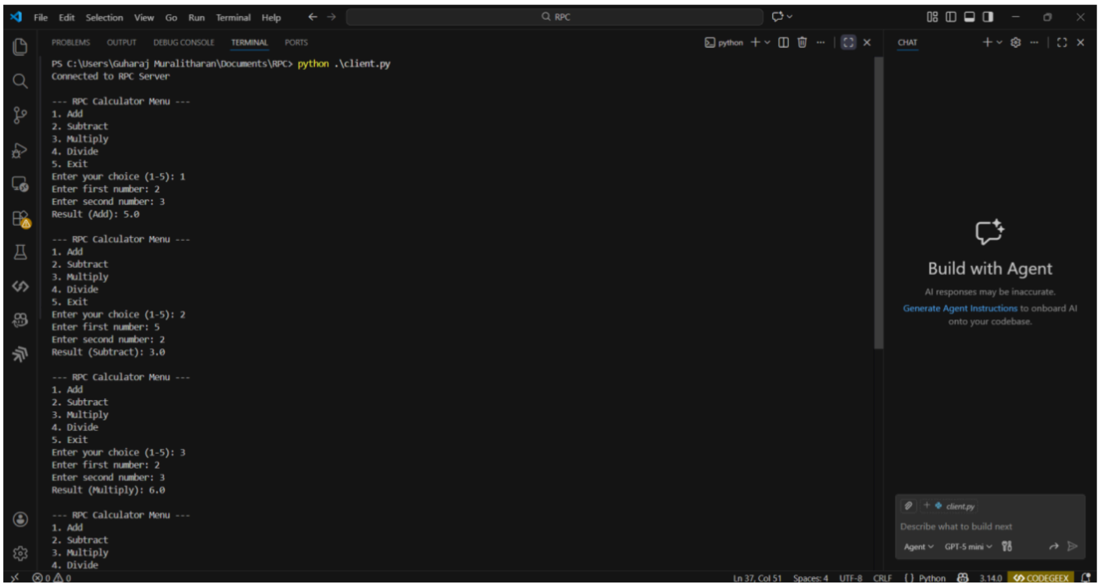
```
┌──(purushoth㉿kali)-[~]
└─$ ssh -i rpc-rmi-key.pem ubuntu@13.60.215.102

Welcome to Ubuntu 22.04.5 LTS (GNU/Linux 6.8.0-1040-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/pro

 System information as of Sun Jan 25 16:38:24 UTC 2026

  System load:  0.08              Processes:             106
  Usage of /:   40.9% of 7.57GB   Users logged in:       0
  Memory usage: 27%               IPv4 address for ens5: 172.31.39.106
  Swap usage:   0%

 * Ubuntu Pro delivers the most comprehensive open source security and
   compliance features.

   https://ubuntu.com/aws/pro

Expanded Security Maintenance for Applications is not enabled.

13 updates can be applied immediately.
To see these additional updates run: apt list --upgradable

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

New release '24.04.3 LTS' available.
Run 'do-release-upgrade' to upgrade to it.


*** System restart required ***
Last login: Sun Jan 25 15:01:39 2026 from 14.139.161.3
ubuntu@ip-172-31-39-106:~$ python3 rpc_server.py
RPC Server running on port 8000
14.139.161.3 - - [25/Jan/2026 16:40:14] "POST /RPC2 HTTP/1.1" 200 -
14.139.161.3 - - [25/Jan/2026 16:40:42] "POST /RPC2 HTTP/1.1" 200 -
14.139.161.3 - - [25/Jan/2026 16:40:52] "POST /RPC2 HTTP/1.1" 200 -
14.139.161.3 - - [25/Jan/2026 16:41:00] "POST /RPC2 HTTP/1.1" 200 -
```

Client invoking remote procedures



```python
from xmlrpc.server import SimpleXMLRPCServer

def add(a, b):
    return a + b

def subtract(a, b):
    return a - b

def multiply(a, b):
    return a * b

def divide(a, b):
    if b == 0:
        return "Error: Division by zero"
    return a / b

server = SimpleXMLRPCServer(("0.0.0.0", 8000))
print("RPC Server running on port 8000")

server.register_function(add)
server.register_function(subtract)
server.register_function(multiply)
server.register_function(divide)

server.serve_forever()
```

- **Correct arithmetic results displayed**

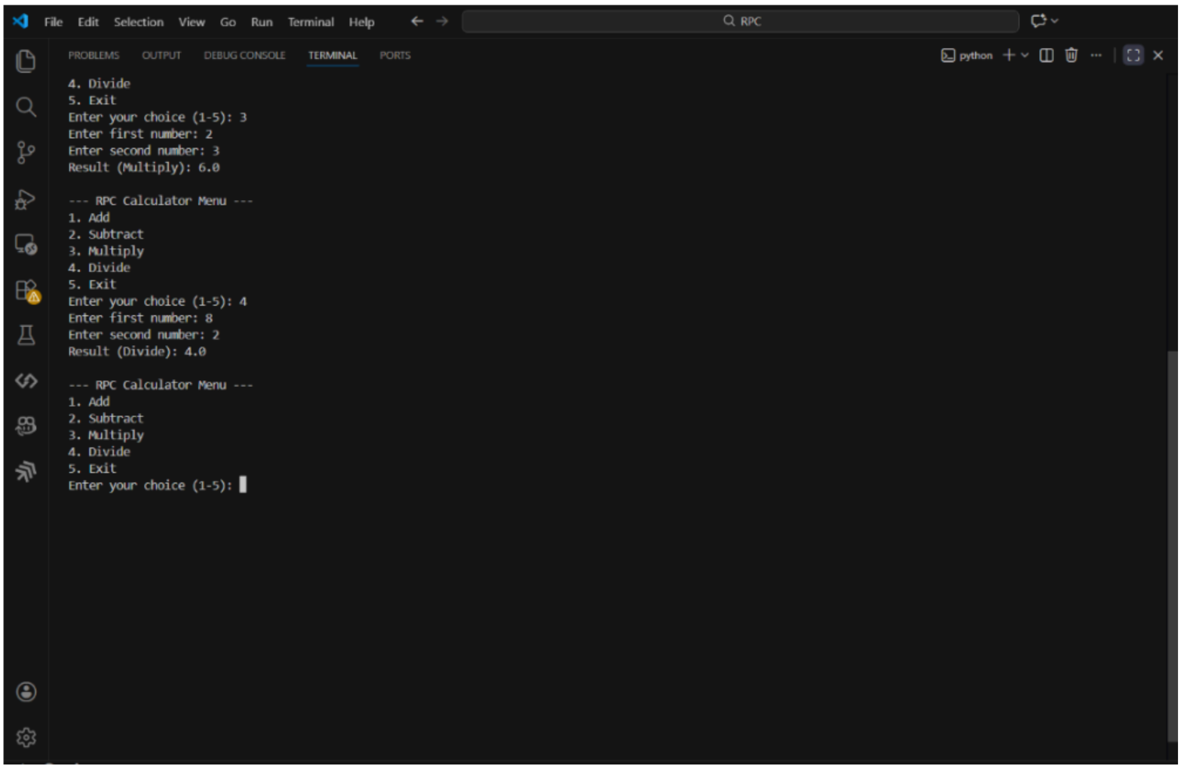


```
PS C:\Users\Guharaj Muralitharan\Documents\RPC> python .\client.py
Connected to RPC Server

--- RPC Calculator Menu ---
1. Add
2. Subtract
3. Multiply
4. Divide
5. Exit
Enter your choice (1-5): 1
Enter first number: 2
Enter second number: 3
Result (Add): 5.0

--- RPC Calculator Menu ---
1. Add
2. Subtract
3. Multiply
4. Divide
5. Exit
Enter your choice (1-5): 2
Enter first number: 5
Enter second number: 2
Result (Subtract): 3.0

--- RPC Calculator Menu ---
1. Add
2. Subtract
3. Multiply
4. Divide
5. Exit
Enter your choice (1-5): 3
Enter first number: 2
Enter second number: 3
Result (Multiply): 6.0

--- RPC Calculator Menu ---
1. Add
2. Subtract
3. Multiply
4. Divide
```

File Edit Selection View Go Run Terminal Help
RPC
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
CHAT
Build with Agent

AI responses may be inaccurate.

Generate Agent Instructions to onboard AI onto your codebase.



```
4. Divide
5. Exit
Enter your choice (1-5): 3
Enter first number: 2
Enter second number: 3
Result (Multiply): 6.0

--- RPC Calculator Menu ---
1. Add
2. Subtract
3. Multiply
4. Divide
5. Exit
Enter your choice (1-5): 4
Enter first number: 8
Enter second number: 2
Result (Divide): 4.0

--- RPC Calculator Menu ---
1. Add
2. Subtract
3. Multiply
4. Divide
5. Exit
Enter your choice (1-5): 
```

File Edit Selection View Go Run Terminal Help
RPC
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
python

9. Result

The RPC-based Calculator Application was successfully implemented and deployed on AWS EC2. The client remotely invoked arithmetic procedures hosted on the cloud server and received accurate computation results.

10. Conclusion

This experiment demonstrated how RPC simplifies distributed computing by allowing remote procedures to be invoked as local functions. Deploying the server on AWS EC2 provided hands-on experience with cloud-based distributed application development.