To deploy this dashboard in a client's account with **least privilege access** (security best practice), you need to set up the infrastructure to allow the dashboard to read data without giving it "Administrator" permissions.

Here is the step-by-step procedure to set up the **EC2 Dashboard Host** and **IAM Permissions** after your Athena tables are ready.

Phase 1: Create the "Least Privilege" IAM Role

Instead of giving full access, we will create a role that can *only* read your specific S3 bucket and run Athena queries.

1. **Go to IAM Console** $\rightarrow$ **Policies** $\rightarrow$ **Create policy**.
2. **Click JSON** and paste this policy (Replace `YOUR-BUCKET-NAME` with the actual bucket name):

   JSON

   ```
   {
       "Version": "2012-10-17",
       "Statement": [
           {
               "Effect": "Allow",
               "Action": [
                   "s3:GetBucketLocation",
                   "s3:GetObject",
                   "s3:ListBucket"
               ],
               "Resource": [
                   "arn:aws:s3:::YOUR-BUCKET-NAME",
                   "arn:aws:s3:::YOUR-BUCKET-NAME/*"
               ]
           },
           {
               "Effect": "Allow",
               "Action": [
                   "s3:GetBucketLocation",
                   "s3:GetObject",
                   "s3:ListBucket",
                   "s3:PutObject"
               ],
               "Resource": [
                   "arn:aws:s3:::YOUR-BUCKET-NAME/athena-results/*"
               ]
           },
           {
               "Effect": "Allow",
               "Action": [
                   "athena:StartQueryExecution",
                   "athena:GetQueryExecution",
                   "athena:GetQueryResults",
                   "athena:StopQueryExecution"
               ],
               "Resource": "*"
           },
           {
               "Effect": "Allow",
   ```

```
                "Action": [
                    "glue:GetTable",
                    "glue:GetDatabase",
                    "glue:GetPartitions"
                ],
                "Resource": "*"
            }
        ]
    }
```

   - *Note: The `PutObject` permission is required for the `athena-results` folder because Athena needs to write the CSV query results there.*

3. **Name the Policy:** `VPC-Dashboard-Policy` and create it.
4. **Go to Roles** $\rightarrow$ **Create role**.
5. **Trusted Entity:** AWS Service $\rightarrow$ **EC2**.
6. **Add Permissions:** Search for and select your new `VPC-Dashboard-Policy`.
7. **Name the Role:** `VPC-Dashboard-Role` and create it.

Phase 2: Launch the Dashboard Server (EC2)

1. **Go to EC2 Console** $\rightarrow$ **Launch Instance**.
2. **Name:** `VPC-Analytics-Dashboard`.
3. **OS Image:** Ubuntu Server 24.04 LTS (Free Tier eligible).
4. **Instance Type:** `t3.micro` (Cheapest/Free Tier).
5. **Key Pair:** Create a new one (e.g., `dashboard-key.pem`) and download it.
6. **Network Settings (Security Group):**
   - Allow **SSH** (Port 22) from **My IP**.
   - Allow **Custom TCP** (Port 8501) from **Anywhere** (0.0.0.0/0) — *This is the dashboard port.*
7. **Advanced Details (Crucial Step):**
   - **IAM Instance Profile:** Select `VPC-Dashboard-Role` (Created in Phase 1).
8. **Launch Instance.**

Phase 3: Install Software on the Server

1. **SSH into the server:**

   Bash

   ```
   chmod 400 dashboard-key.pem
   ssh -i "dashboard-key.pem" ubuntu@<YOUR-EC2-PUBLIC-IP>
   ```

2. **Run the installation commands (copy-paste this block):**

   Bash

   ```
   # 1. Update system
   sudo apt update && sudo apt upgrade -y
   ```

```
# 2. Install Python Environment tools
sudo apt install -y python3-pip python3-venv

# 3. Create a clean environment
python3 -m venv dashboard_env
source dashboard_env/bin/activate

# 4. Install Dashboard Libraries (including the S3 drivers)
pip install streamlit pandas boto3 plotly fsspec s3fs
```

Phase 4: Deploy & Run the Application

1. **Create the file:**

   Bash

   ```
   nano dashboard.py
   ```

2. **Paste** the "Master Code" I provided in the previous response.
   - *Make sure you update the `ATHENA_OUTPUT_BUCKET` variable in the code to match the client's bucket.*
3. **Save & Exit:** `Ctrl+O`, `Enter`, `Ctrl+X`.
4. **Run the Dashboard in Background Mode:**

   (This keeps it running even if you close your laptop)

   Bash

   ```
   nohup streamlit run dashboard.py --server.port 8501 &
   ```

   - *Note: `nohup` stands for "No Hang Up".*

Phase 5: Handover to Client

You can now give the client the URL:

```
http://<EC2-PUBLIC-IP>:8501
```

They will see the full interactive dashboard. They do not need AWS Console access, and the EC2 instance has strictly limited permission to only read the logs it needs.