

Cost Intelligence AI Server for AWS

Simple Launch, Setup, and Validation Guide

AWS Marketplace AMI

Publisher	Code Creator
What this server does	Reads AWS billing exports, checks AWS service connectivity, and queries CUR 2.0 data through Amazon Athena.
Who this guide is for	AWS Marketplace subscribers, cloud administrators, FinOps users, and technical evaluators.
What success looks like	The server starts correctly, accepts setup values, connects to AWS, and returns live Athena billing data.

IMPORTANT: This is a customer guide. It explains how to launch the server, connect the required AWS services, validate the deployment, and use the instance day to day.

What the home page is for

- `http://INSTANCE_URL/` is the front door to the server.
- Before setup, it acts as a welcome page and points you to `/setup`.
- After setup, it serves as the main landing page and dashboard entry point for the product.

Quick start checklist

- Launch the instance with a public IP and open ports 80 and 22 in the security group.
- Attach the EC2 IAM role so the server can use AWS without stored access keys.
- Create a CUR 2.0 Data Export to Amazon S3.
- Create the Athena database and run a Glue crawler on the CUR 2.0 data path.
- Open `/setup`, save your bucket, prefix, database, and workgroup values.
- Recreate the Python containers, then run the validation checks.

IMPORTANT: Do not skip the security group and IAM role steps. Port 80 is needed for the web interface, port 22 is needed for SSH, and the IAM role is needed for AWS access.

1. Before you begin

- Use an AWS account that can create Data Exports, use Amazon S3, run Athena queries, and manage AWS Glue.
- Choose or create an S3 export location such as `s3://your-bucket/exports/`.
- Choose or create an Athena query results location such as `s3://your-bucket/athena-results/`.
- Decide whether you will use the default Athena workgroup named `primary` or a dedicated workgroup.

IMPORTANT: The first CUR 2.0 export file may take time to appear. A new export can take up to 24 hours before the first delivery lands in Amazon S3.

2. Launch the instance and connect

- Launch the AWS Marketplace AMI from AWS Marketplace or the EC2 console.
- Recommended security group: allow inbound HTTP on port 80 and SSH on port 22.
- For testing, t3.large is a good minimum. For steadier use, m6i.large is the safer default.
- Connect by SSH using the ubuntu user and your EC2 key pair and 'ubuntu' user/password
- Attache IAM role to the instance (see below)

IMPORTANT: After launch, check the first-login file before you do anything else. It shows the instance URL, setup URL, and the temporary admin password.

Run these commands after you connect by SSH:

```
cd /opt/codecreator-finops
sudo cat /root/FIRST_LOGIN.txt
sudo systemctl status finops-firstboot.service --no-pager
docker compose ps
curl http://localhost/health
curl http://localhost/api/health
```

- Open the server in your browser at http://PUBLIC_IP/
- Open the setup page at http://PUBLIC_IP/setup

3. Attach the IAM role (IMPORTANT)

IMPORTANT: Use an EC2 IAM role instead of storing AWS access keys on the server. The application is designed to use temporary credentials from the attached instance role.

- If you are launching a new instance, choose the role during launch.
- **If the instance is already running, go to EC2 → Instances → select the instance → Actions → Security → Modify IAM role.**
- The role should allow Cost Explorer reads, S3 read access for the export bucket, S3 write access for Athena query results, Glue metadata reads, and Athena query execution.

4. Create the CUR 2.0 export

- Open Billing and Cost Management in the AWS console.
- Go to Data Exports and choose Create export.
- Choose Standard data export, then choose CUR 2.0.
- Give the export a name such as finops-cur2.
- Choose your S3 bucket and use a prefix such as exports/.
- Choose Parquet output.
- Use overwrite mode for Athena-friendly delivery.
- Save the export and allow AWS to apply the S3 bucket policy if prompted.

IMPORTANT: Do not assume the export is broken if the bucket is empty right away. Wait for the first file to appear before moving to Athena validation.

5. Prepare Athena and AWS Glue

- Open Amazon Athena in the same AWS Region as the instance and export bucket.
- If Athena asks for a query results location, set one first.
- Run CREATE DATABASE finops; in the Athena Query Editor.
- Create an AWS Glue crawler that points to the export data path, not the metadata path.
- Use a dedicated AWS Glue role whose trust relationship allows glue.amazonaws.com.
- Set the crawler target database to finops and run the crawler.
- Confirm that a table appears in the finops database.

Useful example values

Typical export data path: `s3://your-bucket/exports/finops-cur2/data/`

Athena SQL: `CREATE DATABASE finops;`

Athena SQL: `SHOW TABLES IN finops;`

6. Configure the instance

- Preferred method: open `http://PUBLIC_IP/setup` and enter your values in the setup page.
- Enter your admin email, S3 export bucket, S3 prefix, Athena database, and Athena workgroup.
- If your build still shows a placeholder setup page, use the API method below instead.

API-based setup method

```
curl -X POST http://localhost/api/setup \  
-H "Content-Type: application/json" \  
-d '{  
  "admin_email": "you@example.com",  
  "s3_export_bucket": "your-export-bucket",  
  "s3_export_prefix": "exports/",  
  "athena_database": "finops",  
  "athena_workgroup": "primary",  
  "billing_view_arn": "",  
  "default_cost_category": "Team",  
  "bedrock_enabled": false,  
  "bedrock_model_id": "",  
  "public_base_url": "http://PUBLIC_IP"  
}'
```

IMPORTANT: After setup, recreate the Python containers.

```
cd /opt/codecreator-finops
```

```
sudo docker compose up -d --force-recreate api worker beat
```

7. Validate the deployment

AWS connectivity check

```
curl http://localhost/api/aws/check
```

- sts.ok should be true.
- cost_explorer.ok should be true.
- s3.ok should be true.
- glue.ok should be true.

Athena checks through the application

```
curl http://localhost/api/athena/count
```

```
curl http://localhost/api/athena/sample
```

- The count endpoint should return a numeric row count.
- The sample endpoint should return CUR 2.0 columns and live values.

IMPORTANT: If the Athena endpoints fail after setup, confirm that the role can run Athena queries and write Athena query results to S3.

8. How to use the server day to day

- Open http://PUBLIC_IP/ to reach the main landing page.
- Open http://PUBLIC_IP/setup if you need to update the export bucket, prefix, Athena database, or Athena workgroup.
- Use `/api/aws/check` when you want a quick AWS connectivity test.
- Use `/api/athena/count` and `/api/athena/sample` when you want to confirm that billing data is still queryable.
- Use `/root/FIRST_LOGIN.txt` as the first place to look for the initial URL and password after launch.

9. Useful locations and endpoints

Path or endpoint	Purpose
<code>/opt/codecreator-finops</code>	Main application directory.
<code>/opt/codecreator-finops/runtime/customer.env</code>	Customer-specific runtime settings after setup.
<code>/root/FIRST_LOGIN.txt</code>	First-boot summary for the instance owner.
<code>http://PUBLIC_IP/</code>	Main landing page and dashboard entry point.

http://PUBLIC_IP/setup	Setup page.
/api/aws/check	AWS connectivity check.
/api/athena/count	Athena row count check.
/api/athena/sample	Athena sample result check.

10. Troubleshooting

- **The export bucket stays empty** — Confirm that the Data Export points to the expected bucket and prefix. The first delivery may take time.
- **The Glue crawler cannot be created** — Use a dedicated AWS Glue role and confirm that the trust relationship allows glue.amazonaws.com.
- **Athena says the database does not exist** — Create the database first in Athena, then rerun the crawler.
- **The app can reach AWS but Athena queries fail** — Confirm that the instance role can run Athena queries and write Athena query results to S3.
- **The app still shows old values** — Recreate the API, worker, and beat containers after changing runtime settings.
- **The setup page is only a placeholder** — Use the API-based setup method in this guide.

11. Security notes

IMPORTANT: Keep the export bucket private and leave Block Public Access enabled unless you have a specific reason to change it.

- Use an IAM role attached to the instance instead of storing AWS access keys on disk.
- Limit inbound access on ports 22 and 80 to the networks you actually need.
- Remove customer-specific runtime settings and temporary files before creating a new Marketplace AMI.