**TASK17- COST OPTIMIZING BY CLEANING UP STALE RESOURCES**

The **CLEANUP\_ORPHANED\_EBS\_RESOURCES** task is designed to automatically identify and delete unused Amazon Elastic Block Store (EBS) resources. This includes both EBS volumes that are no longer attached to any running EC2 instances and EBS snapshots that are no longer associated with any volume. The task helps in cleaning up orphaned resources to optimize AWS cost and resource management by ensuring that stale and unused volumes and snapshots are promptly deleted.

**Key Features:**

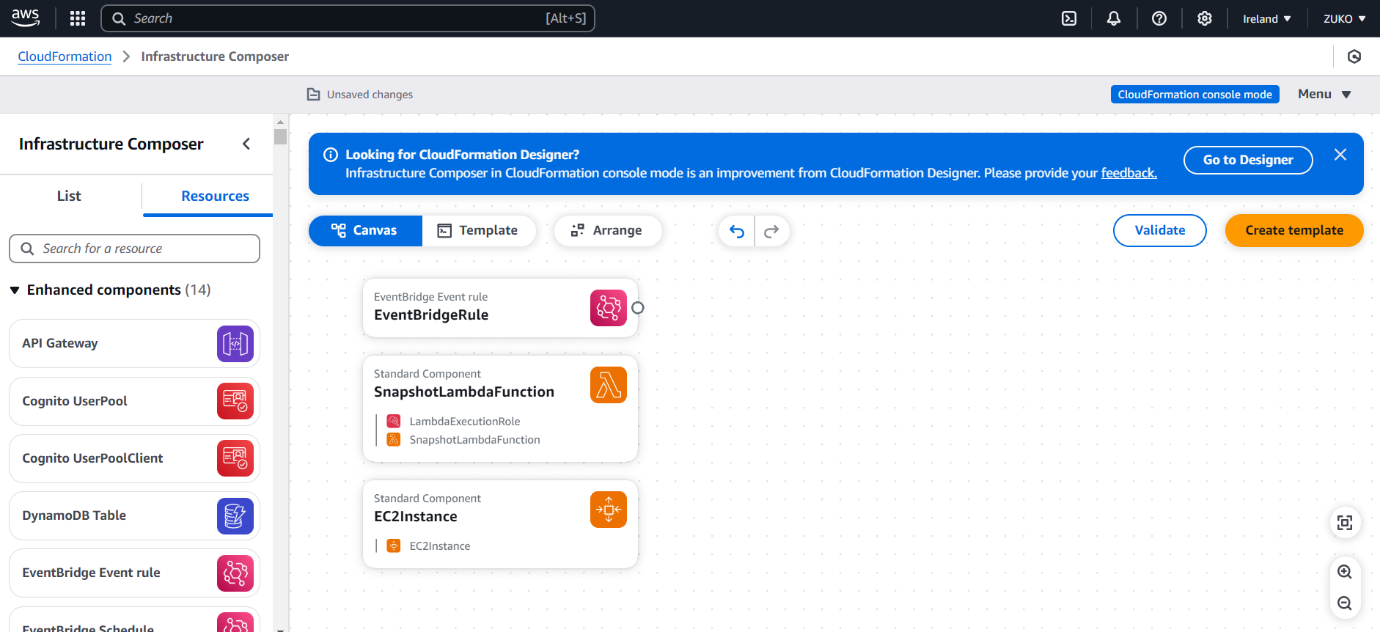
* **Detects Stale EBS Volumes:** Identifies and deletes EBS volumes that are in the "available" state, meaning they are no longer attached to any running EC2 instances.
* **Deletes Orphaned Snapshots:** Finds and deletes snapshots that are no longer associated with any active EBS volumes.
* **Detaches Volumes from Stopped Instances:** If volumes are attached to stopped instances, they are first detached before being deleted.
* **Efficient Resource Management:** Helps reduce unnecessary storage costs by ensuring unused resources do not accumulate over time.

**Workflow:**

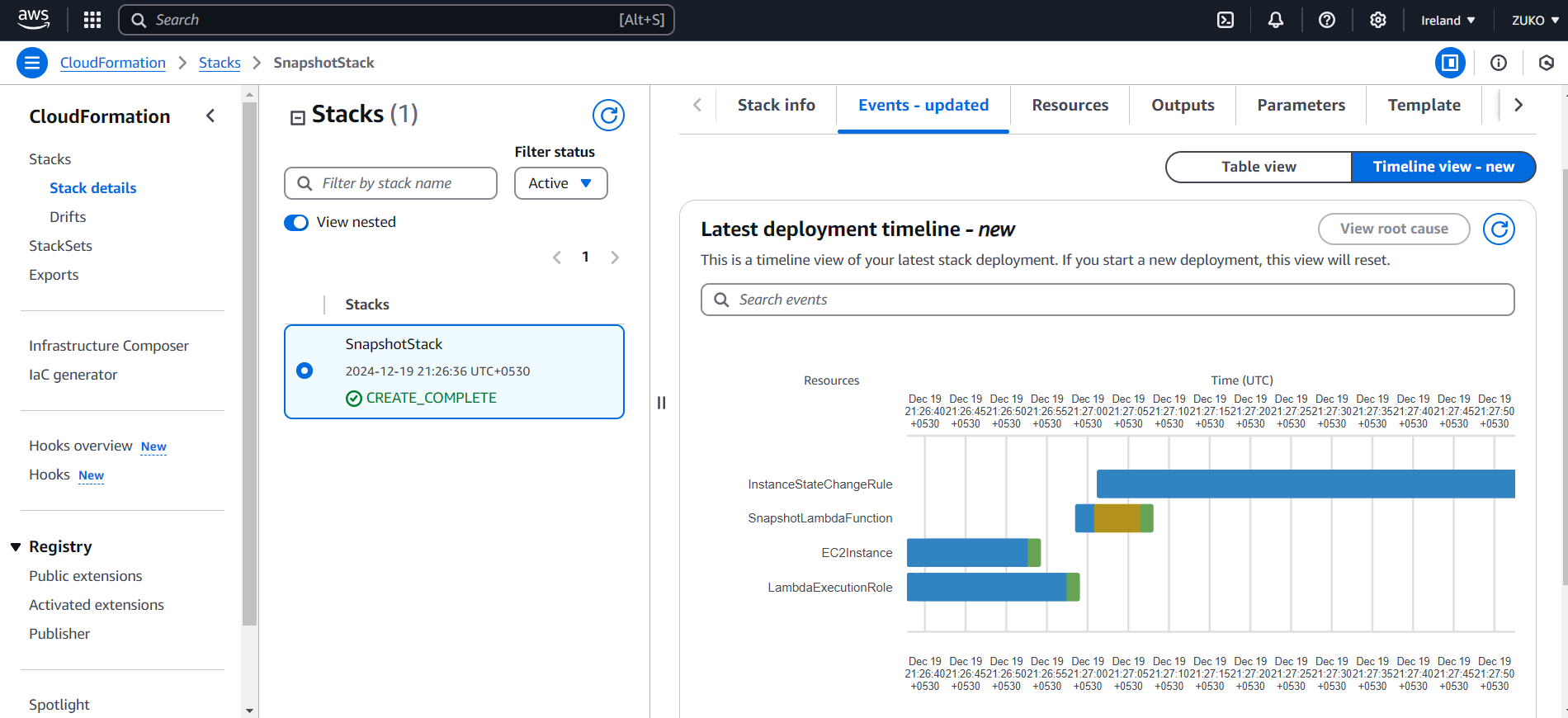
1. The Lambda function can be triggered manually or by an event (e.g., EC2 instance termination or periodic schedule).
2. It identifies volumes attached to terminated or stopped instances.
3. The function detaches any volumes still attached to stopped instances and deletes them.
4. It identifies snapshots associated with orphaned volumes or without any associated volume.
5. Deletes any snapshots identified as orphaned.
6. Logs the actions performed and provides confirmation once the cleanup process is completed.

**Use Case:**  
This task is ideal for use in environments with frequent EC2 instance termination or volume snapshots, helping maintain a clean and cost-effective AWS environment by removing resources that are no longer needed.

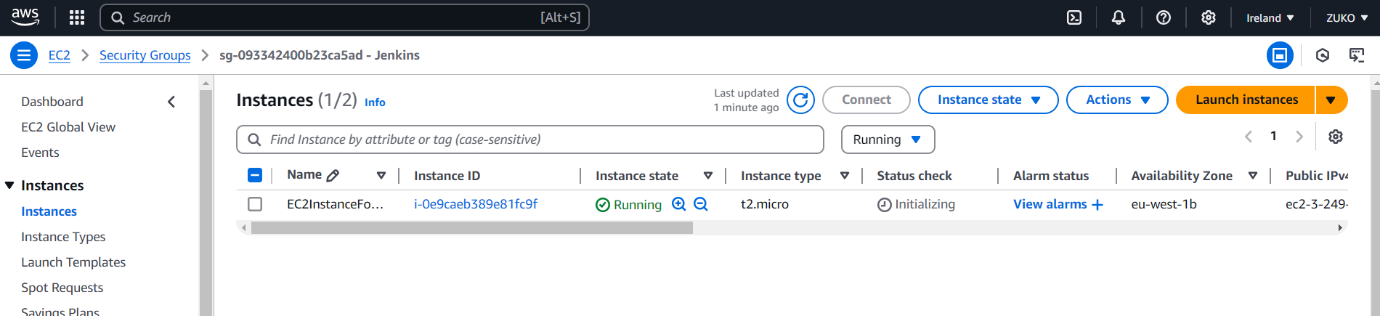
USING AWS CFT TO BUILD INFRA USING CODE (YAML)



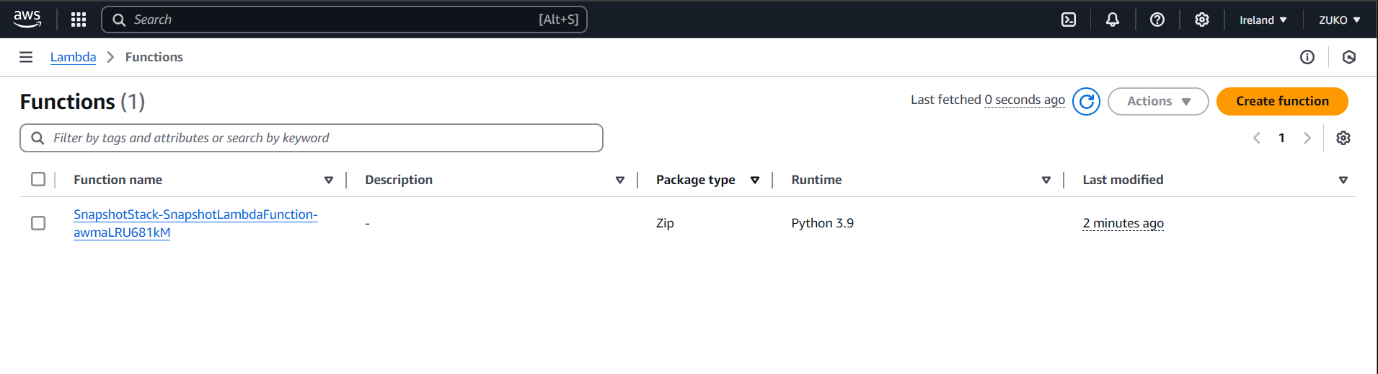
STACK CREATED SUCCESSFULLY



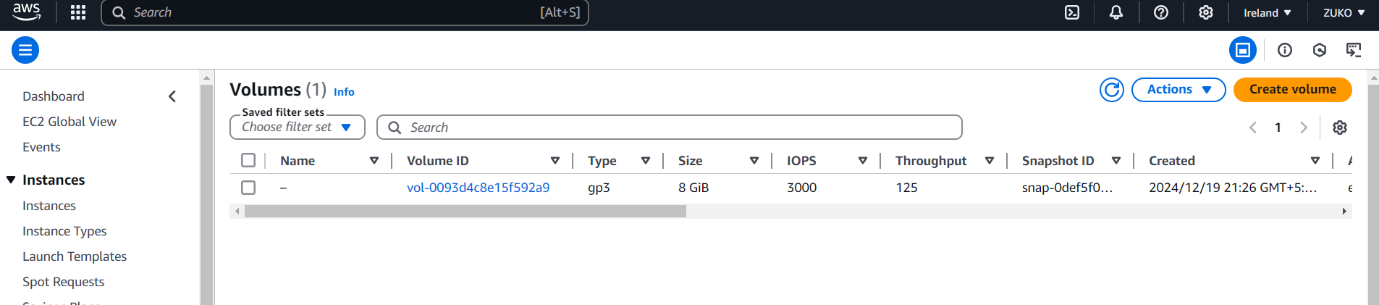
EC2 INSTANCE CREATED USING IAAC



USING LAMBDA FUNCTION TO DETECT AND DELETE STALE RESOURCES

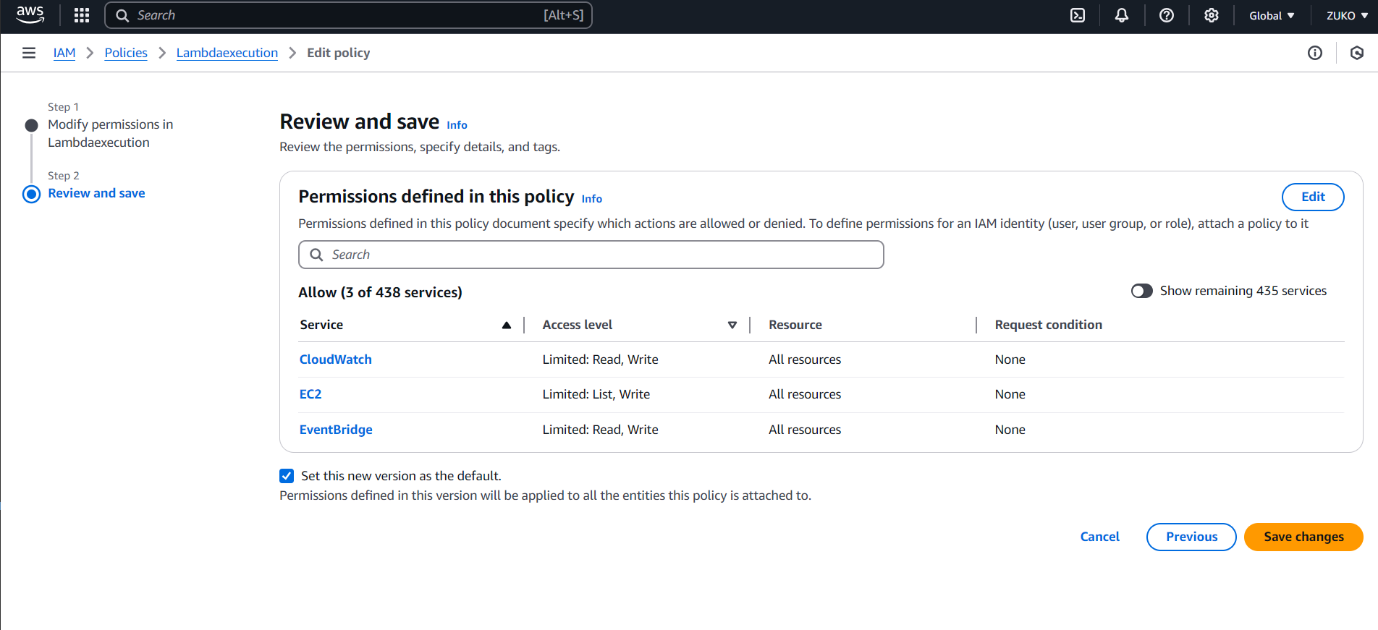


EBS VOLUME CREATED AND ATTACHED USING IAAC

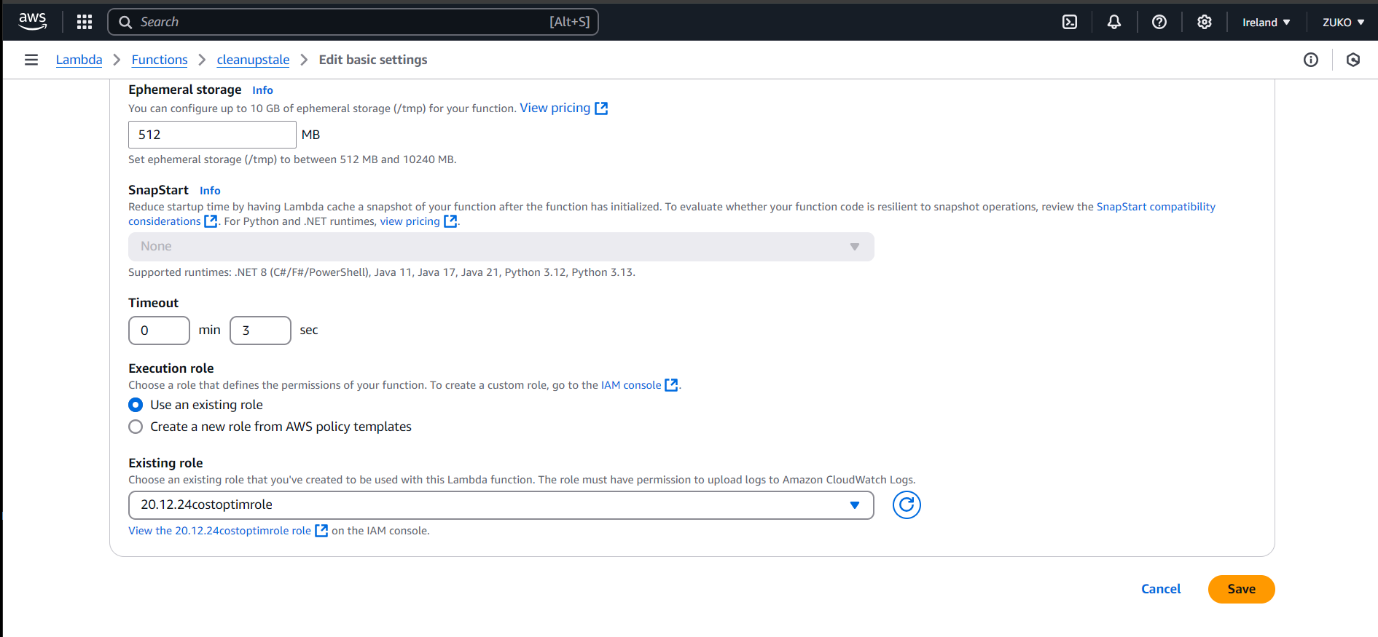


SNAPSHOTS CANNOT BE CREATED USING CFT, WE CAN USE LAMBDA FUNCTION TO CREATE SNAPSHOT

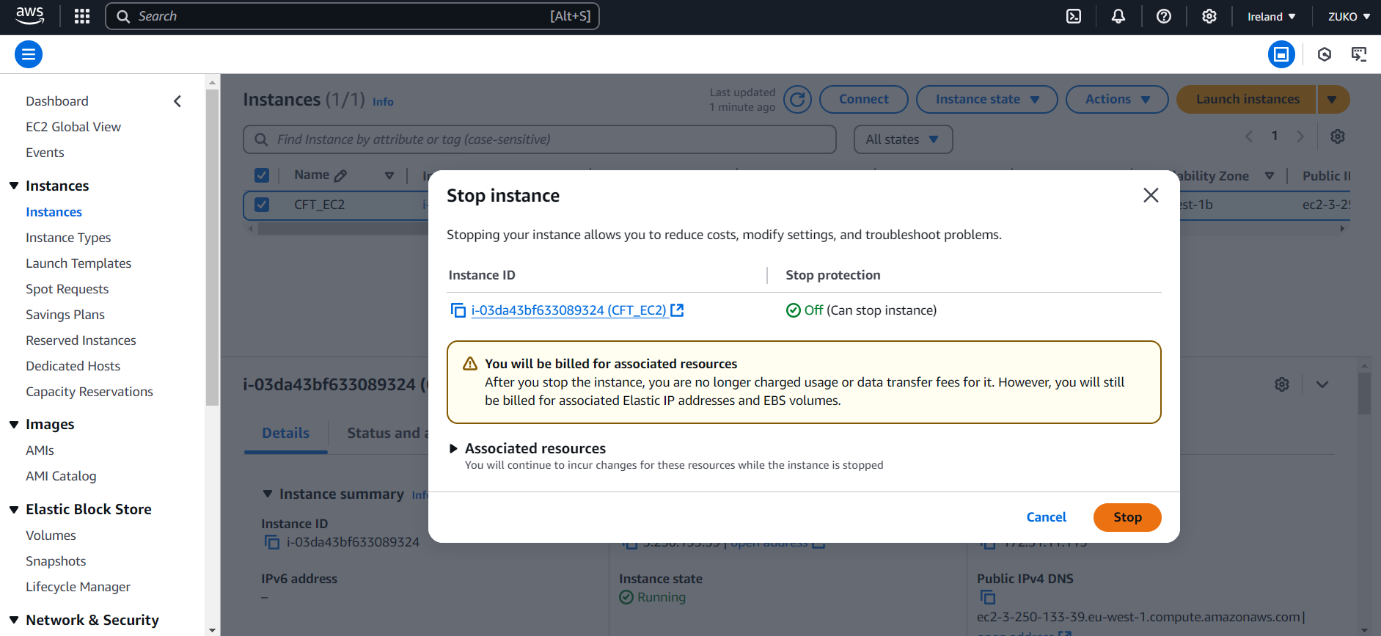
PROVIDING PERMISSION TO LAMBDA VIA IAM ROLE



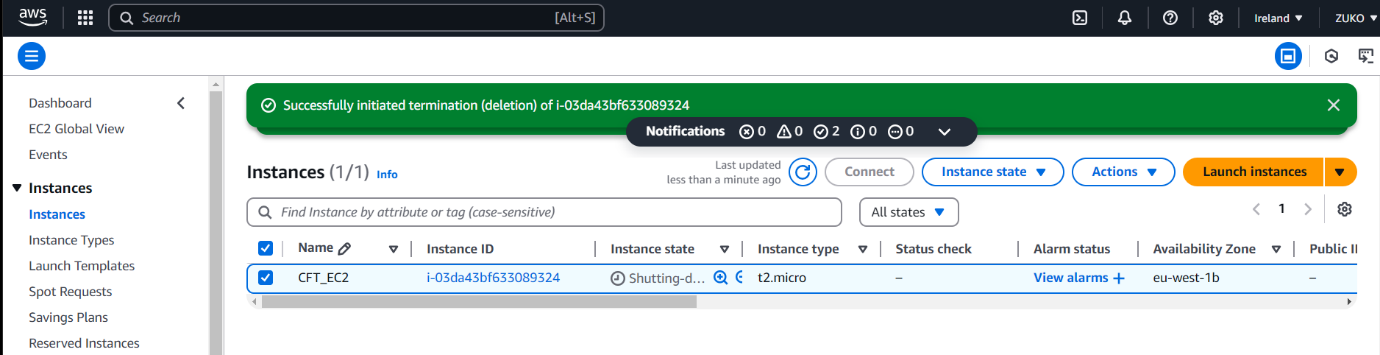
ATTACHING CUSTOMIZED IAM ROLE TO LAMBDA FUNCTION



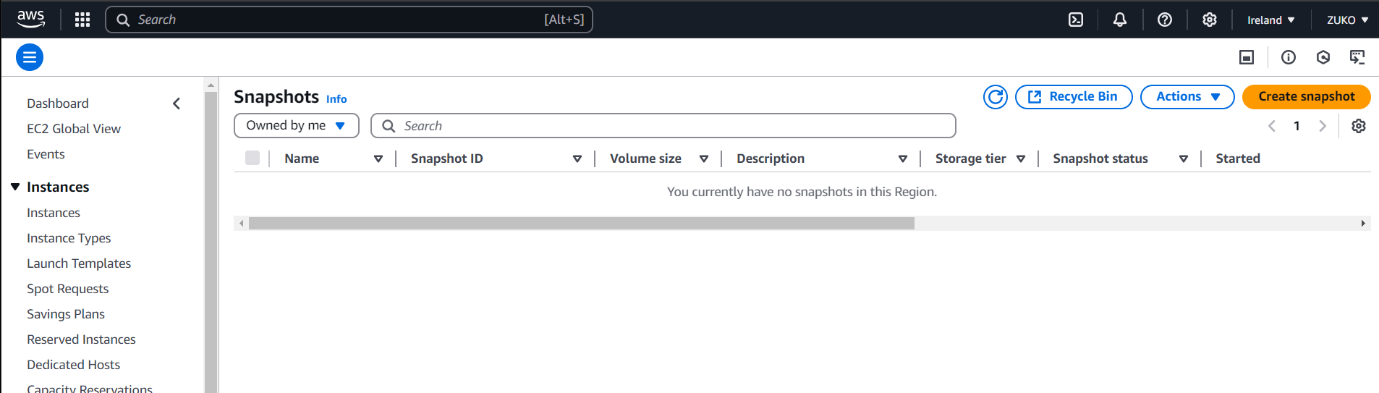
STOPPING INSTANCE MANUALLY



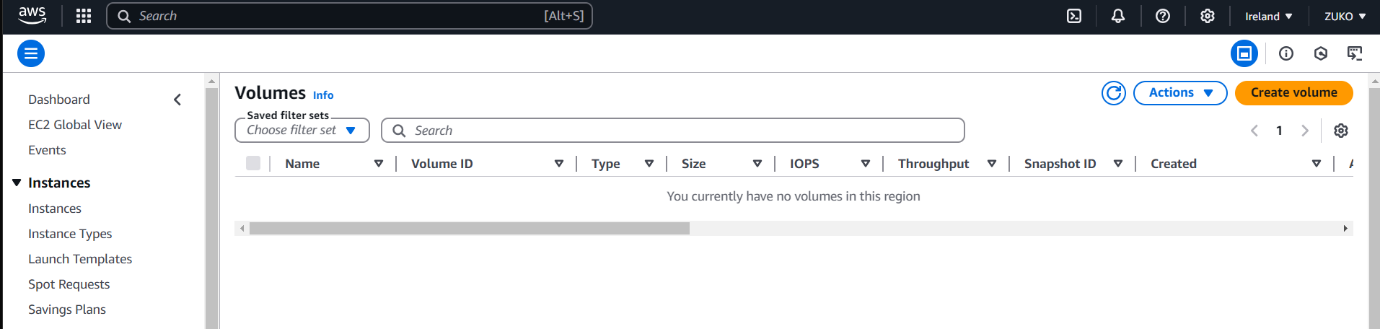
TERMINATING INSTANCE MANUALLY



LAMBDA DETECTED AND DELETED STALE SNAPSHOT



EBS VOLUME GOT DELETED

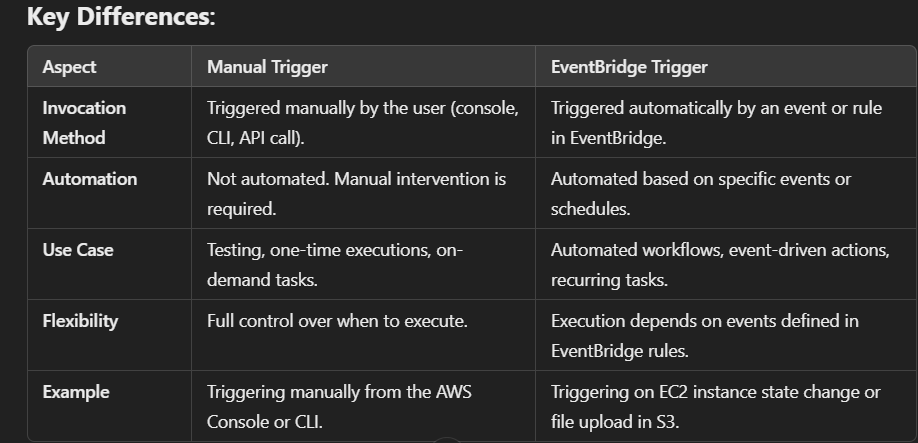


The primary difference between **manually triggering a Lambda function** and **using EventBridge** to trigger the function lies in how and when the Lambda function is invoked, and how it integrates with other AWS services.

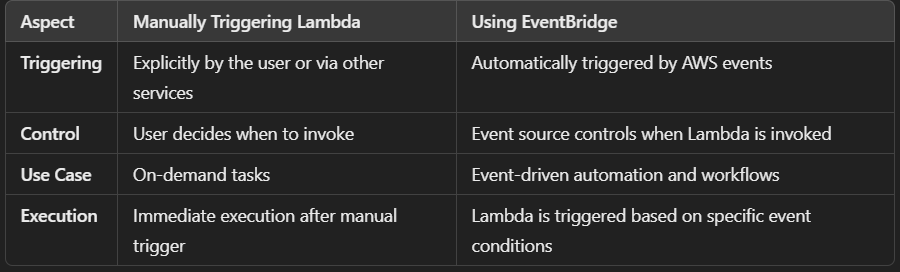
**1. Manually Triggering a Lambda Function**

* **Definition**: Manually triggering a Lambda function means you invoke the function directly, typically from the AWS Lambda Console, the AWS CLI, or an API call.
* **Use Cases**:
  + When you need to run the Lambda function on demand for specific tasks or testing purposes.
  + You have precise control over when the Lambda function is executed.
  + Suitable for troubleshooting, testing, or executing Lambda functions based on immediate needs.
* **How it Works**:
  + You initiate the function yourself, either by clicking the **Test** button in the Lambda Console, using the AWS CLI with aws lambda invoke, or through an API Gateway call.
  + You provide the input data manually (through the console or API).
* **Pros**:
  + Immediate, on-demand execution.
  + You can test the Lambda function before setting up automation.
  + Full control over when to invoke the Lambda function.
* **Cons**:
  + No automatic trigger; you must remember to manually invoke the function.
  + Not suitable for automated or recurring tasks.

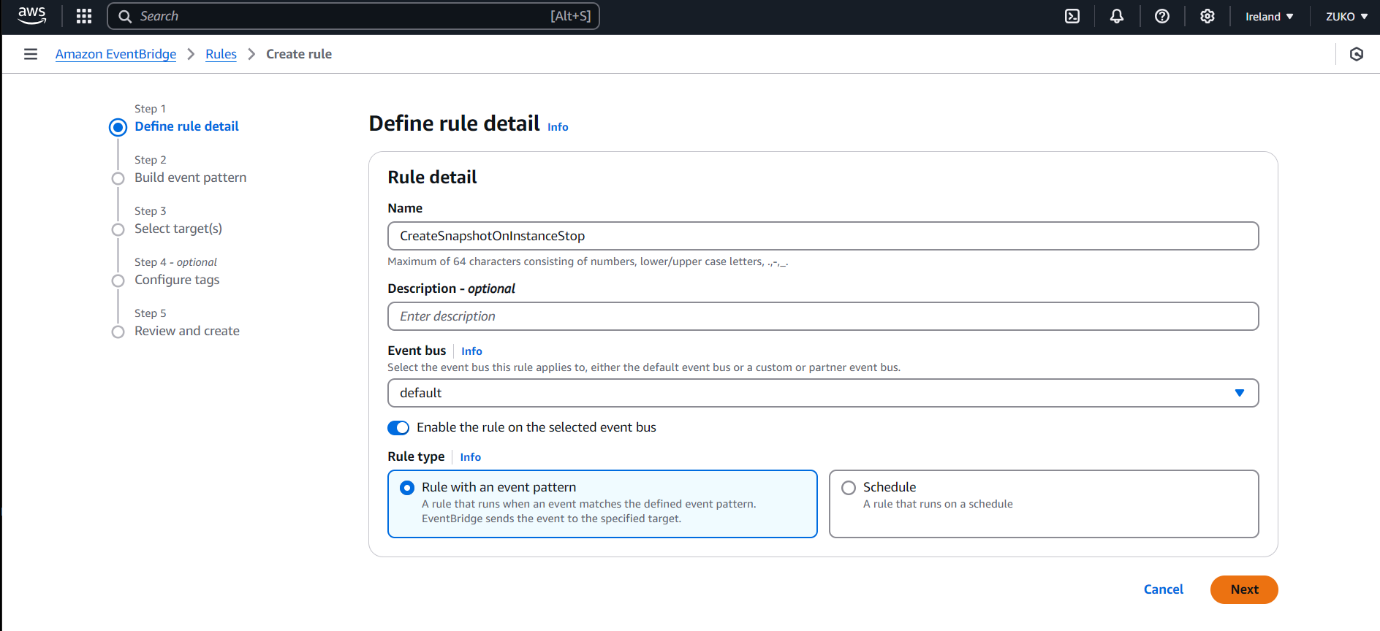
**2. Using EventBridge to Trigger a Lambda Function**

* **Definition**: Amazon EventBridge is a fully managed event bus that allows you to set up automated workflows by triggering Lambda functions based on events happening within AWS services or custom events.
* **Use Cases**:
  + Automatically invoking a Lambda function based on specific events (e.g., EC2 instance state changes, S3 file uploads, CloudWatch alarms).
  + Automating tasks like resource management, event-driven architecture, or reacting to changes in your AWS environment.
  + Scheduling Lambda functions for recurring tasks (e.g., cleaning up resources every day at midnight).
* **How it Works**:
  + You define a rule in EventBridge to match certain events (e.g., EC2 instance state change, S3 object creation, etc.).
  + When that event occurs, EventBridge automatically triggers the Lambda function with the event data passed to it.
  + You don't need to invoke the function manually. The event triggers the function automatically based on the defined conditions.
* **Pros**:
  + Automated execution based on specific events or schedules.
  + Ideal for building event-driven architectures.
  + No need for manual intervention; the function is triggered automatically as part of the system.
* **Cons**:
  + Requires configuring EventBridge rules and setting up triggers.
  + Limited control over the exact moment of execution (you rely on the event happening).
* 

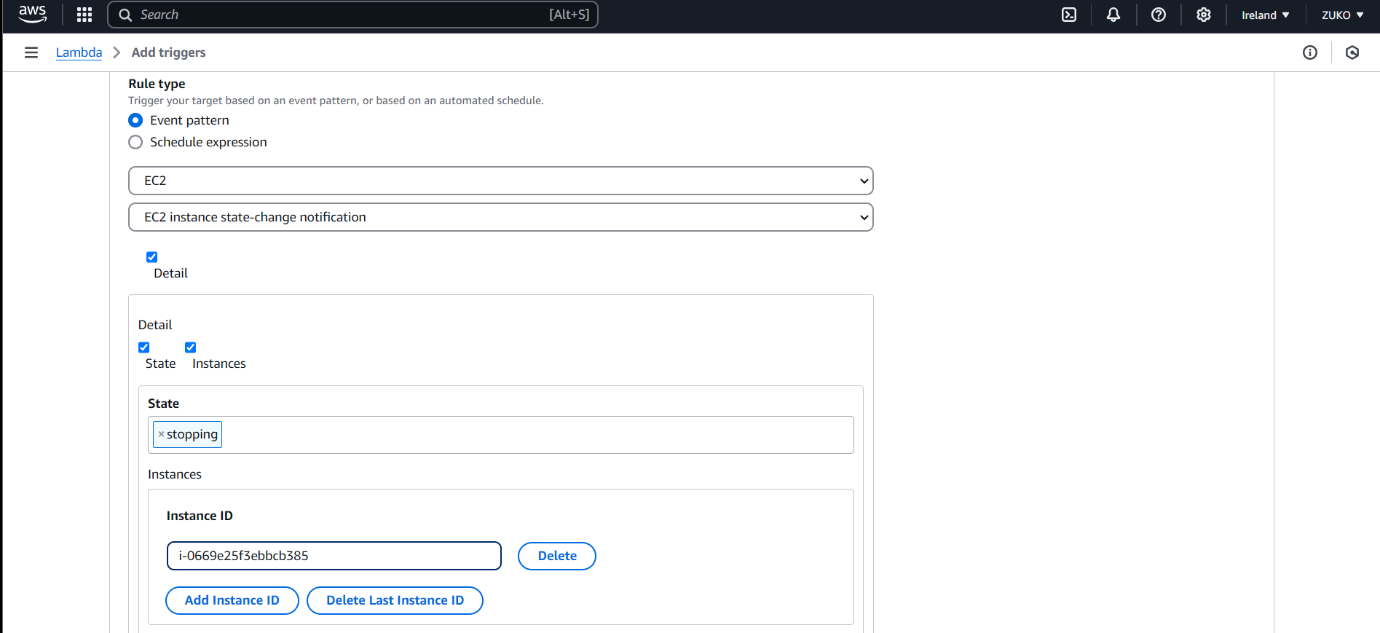
**When to Use Each:**

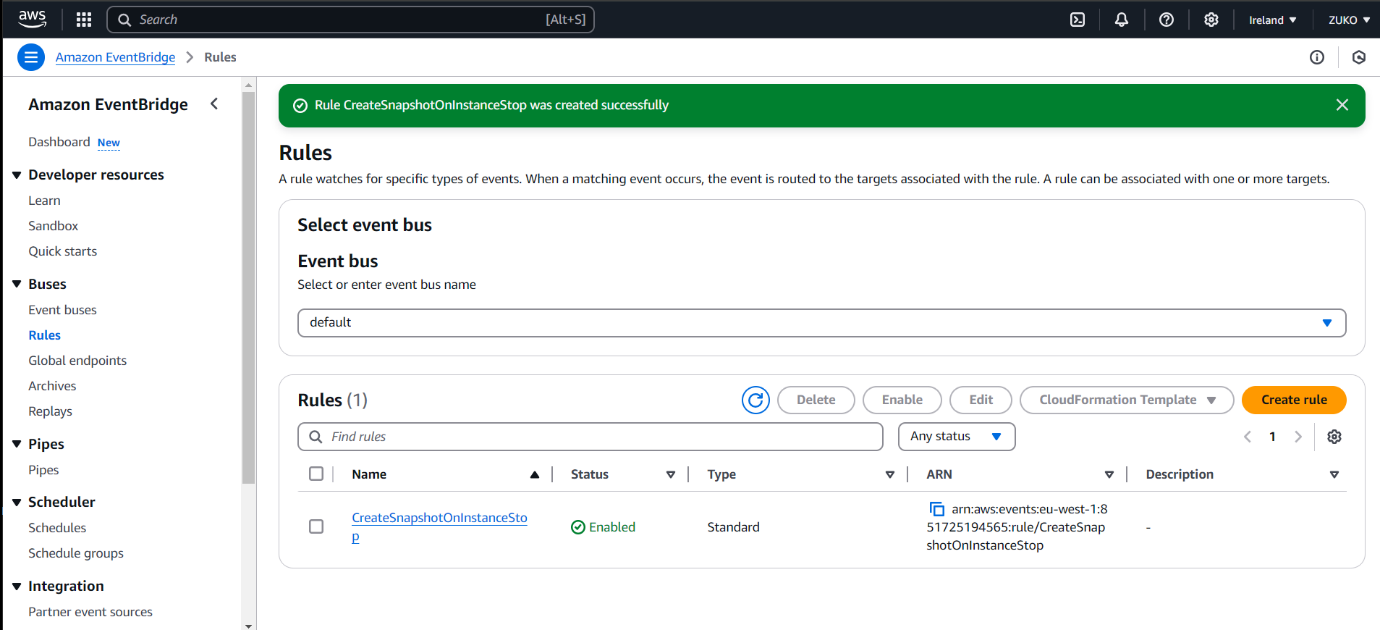
* **Manually Triggered Lambda**: Use this when you want full control over the execution, such as in testing or debugging scenarios.
* **Event Bridge Triggered Lambda**: Use this when you need automation and event-driven workflows, where the Lambda function should execute automatically based on
* AWS events or scheduled time.
* 

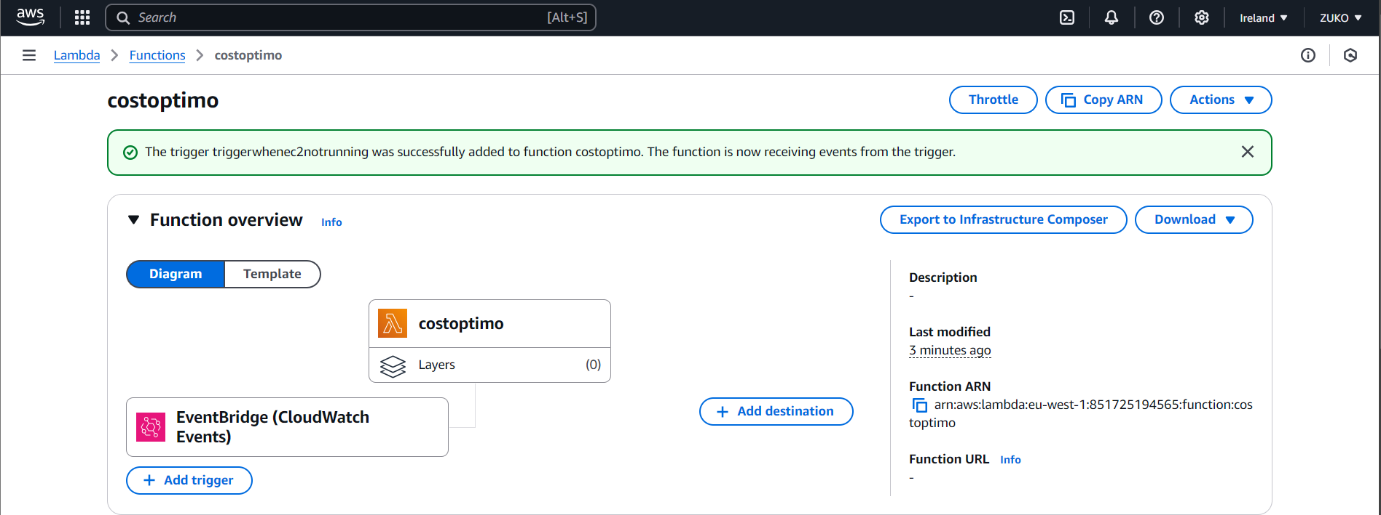
TRIGGERING LAMBDA USING EVENT PATTERN

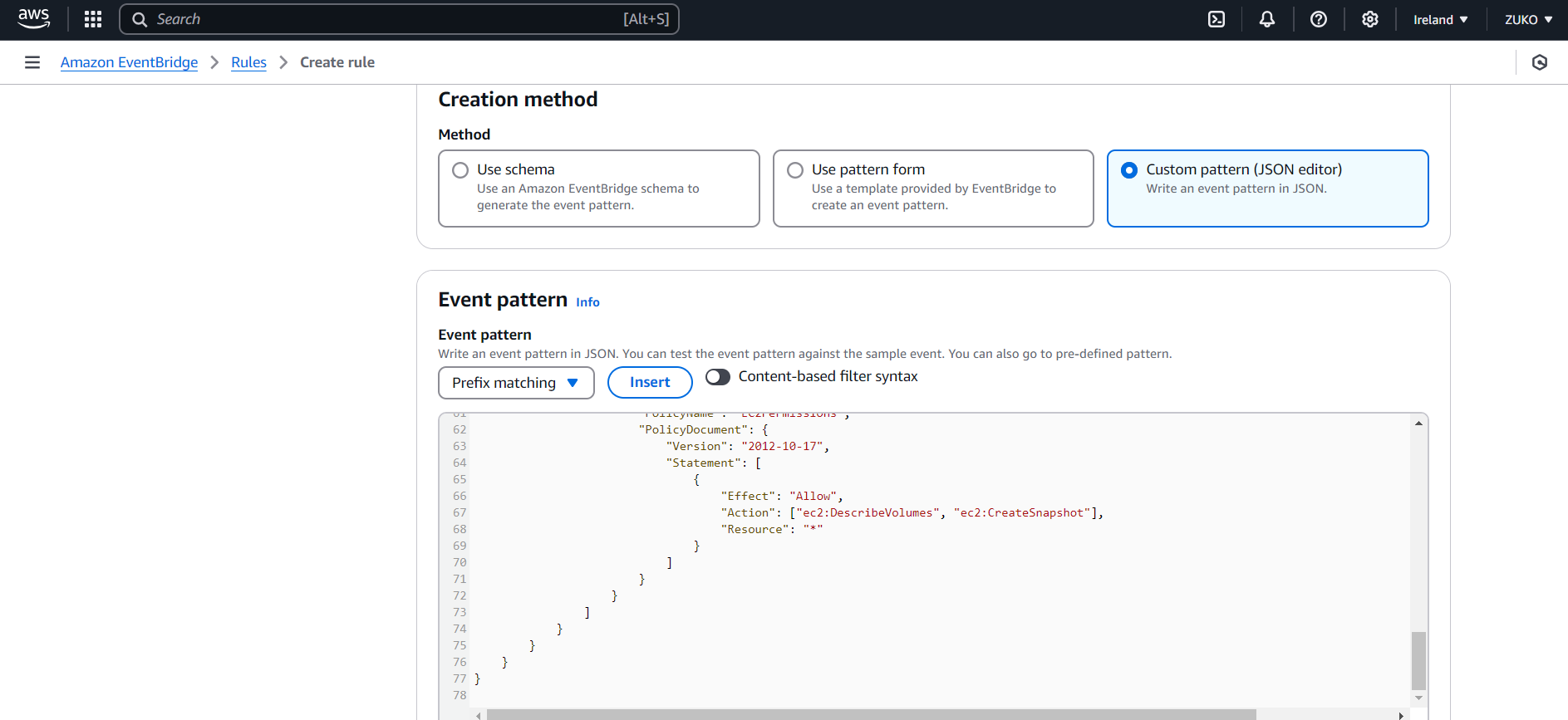


PROVIDING INSTANCE DETAILS

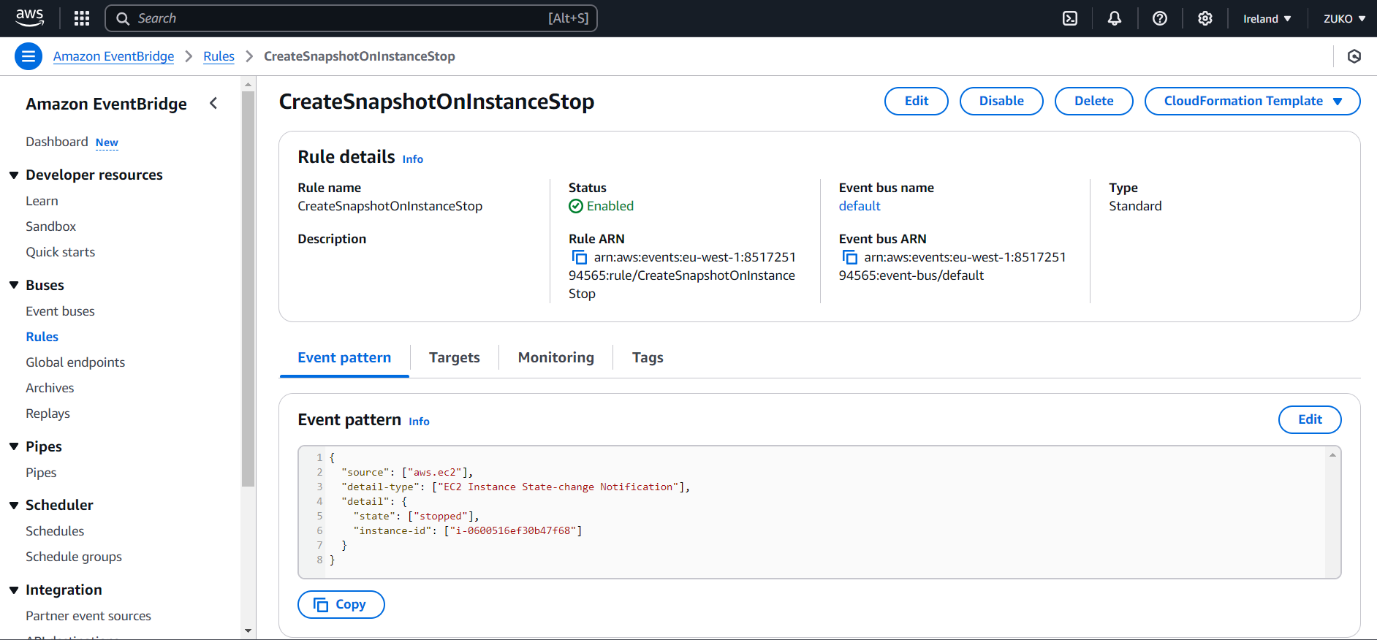




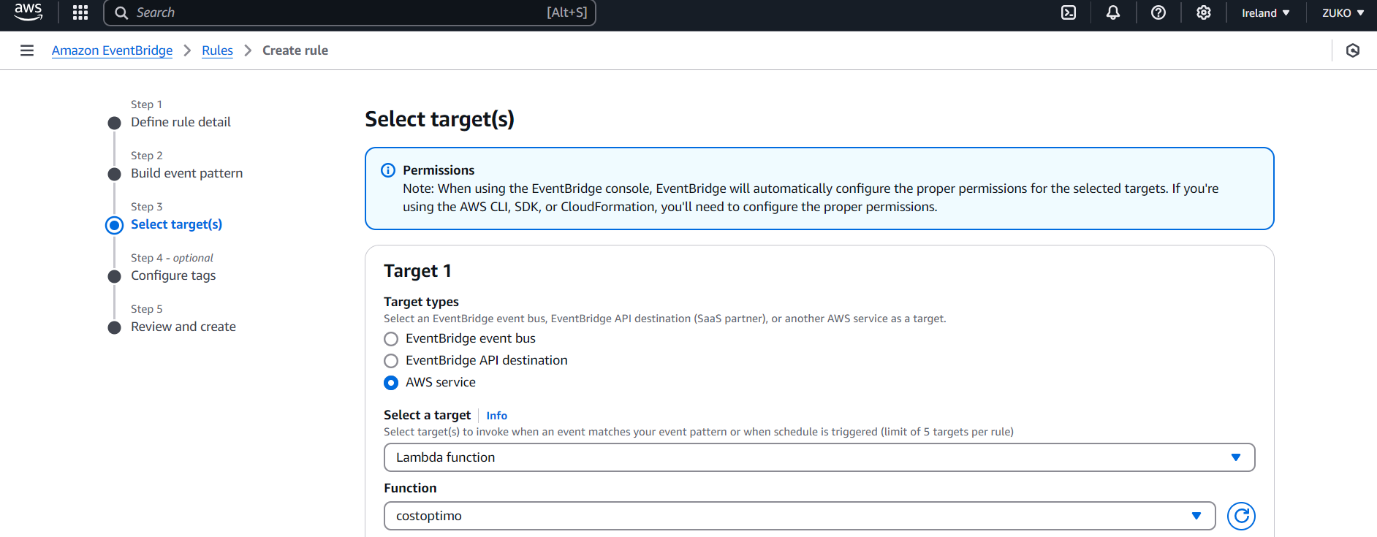




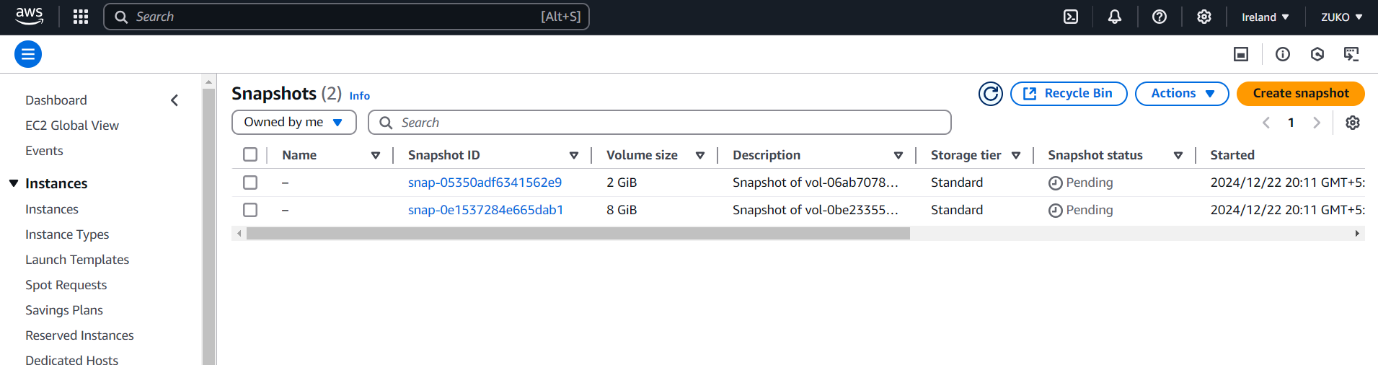
TRIGGERING SNAPSHOT CREATION USING EVENT BRIDGE



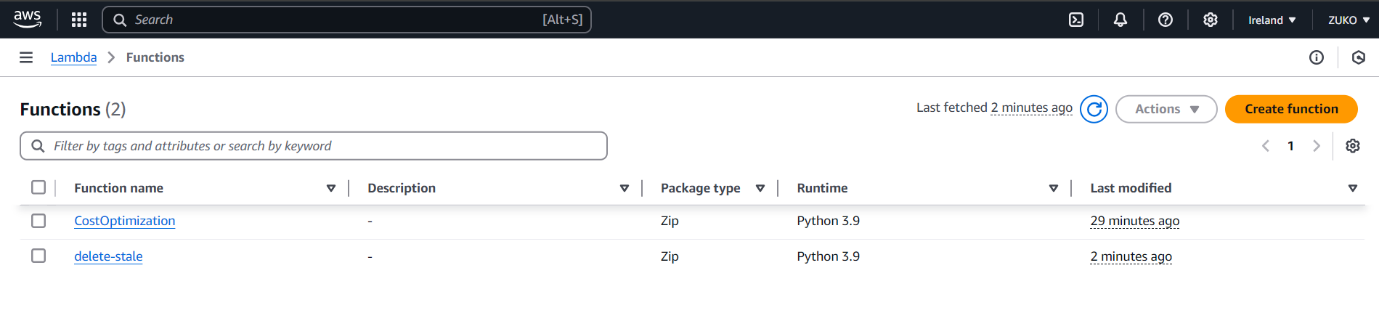
PROCEEDING WITH TARGETS



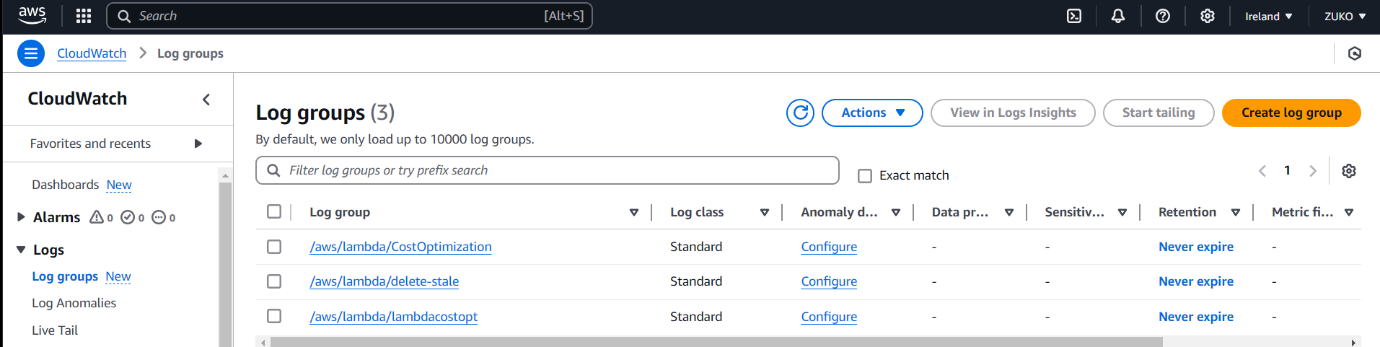
SNAPSHOTS CREATED FOR EBS VOLUMES & INSTANCE ROOT VOLUME



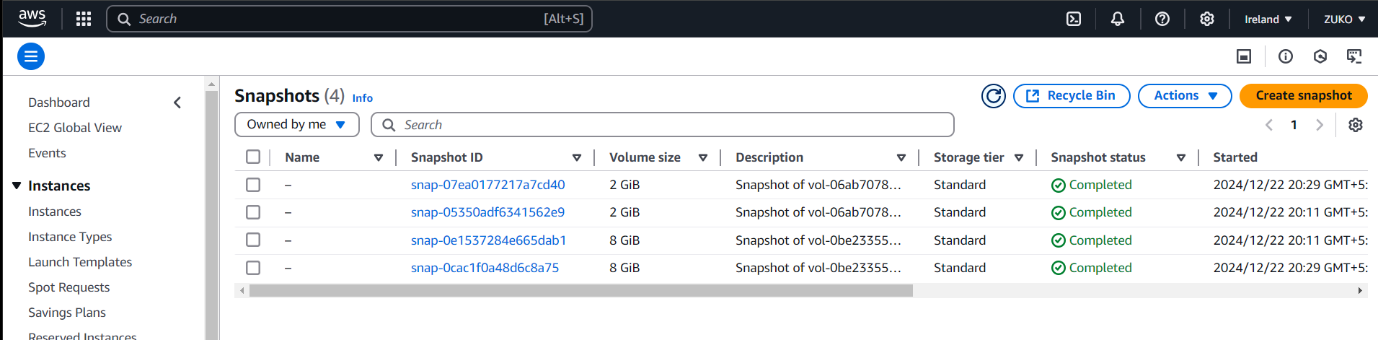
LAMBDA FUNCTIONS



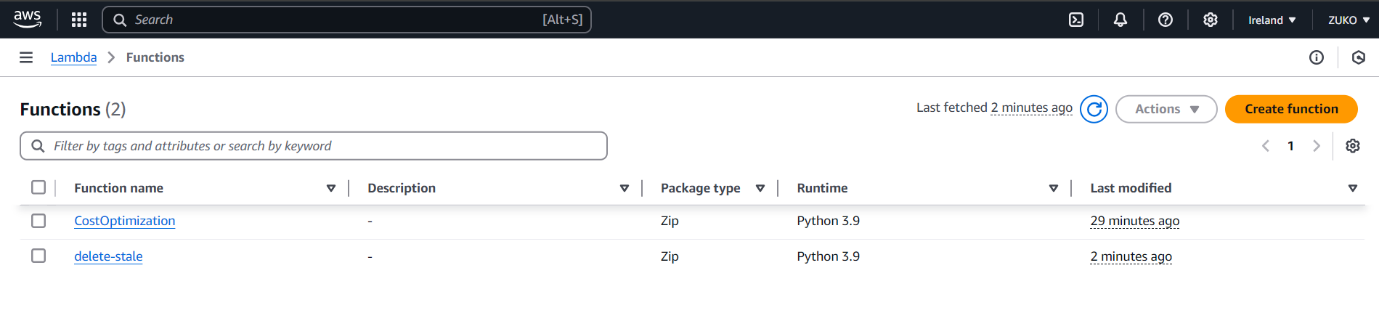
CLOUDWATCH LOG GROUPS



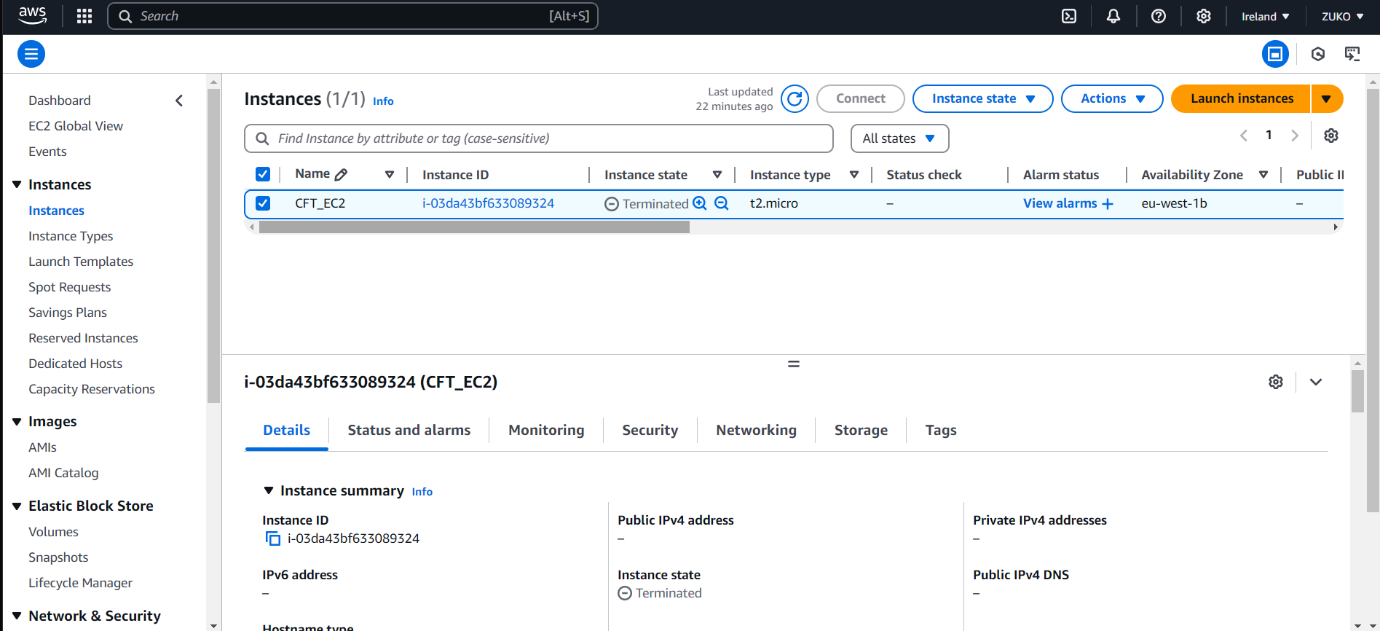
STOPPED THE INSTANCE TWICE, SNAPSHOT CREATED TWICE



NEW LAMBDA FUNCTION TO DELETE STALE RESOURCES



DELETING INSTANCE MANUALLY



STALE RESOURCE DELETED

