ECS Tasks Definitions

IAM Role

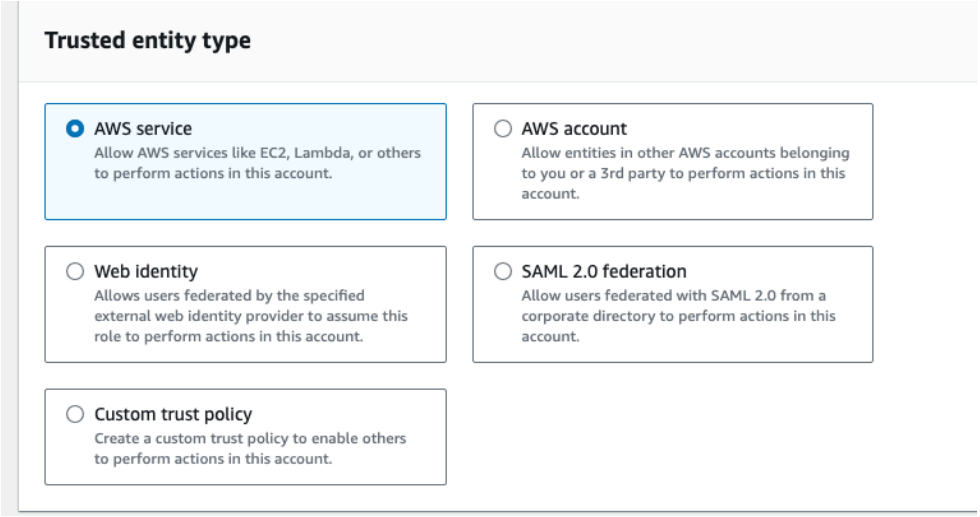
ECS Fargate is the serverless container orchestrator. It needs 2 IAM roles.

1. A role for our netflux application to call ***secrets manager*** to get the RDS credentials.
2. A role for Fargate cluster to pull the docker image and run the application.

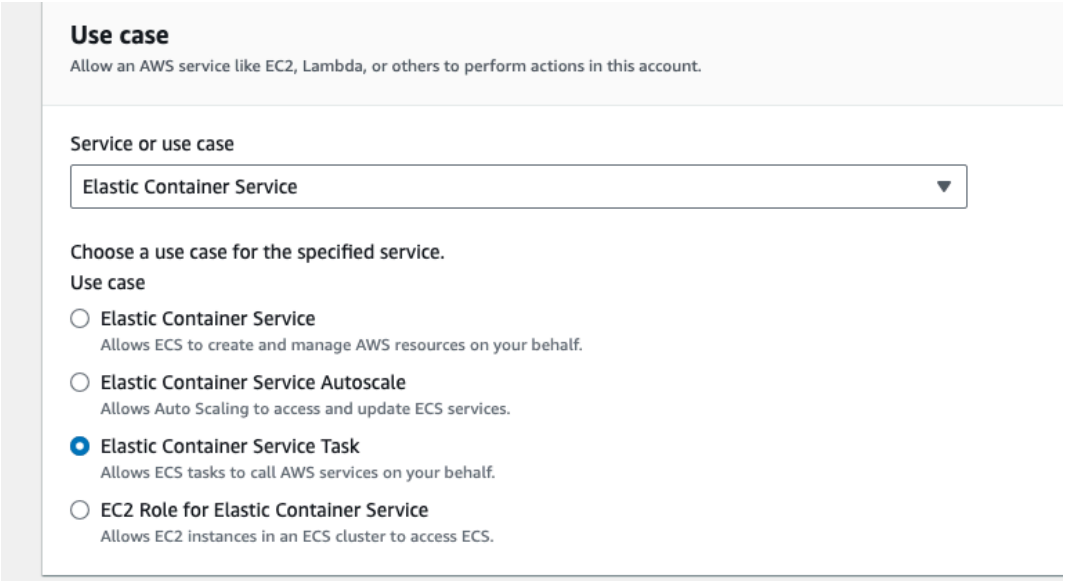
Let's create these one by one.

Role 1: netflux-task-role

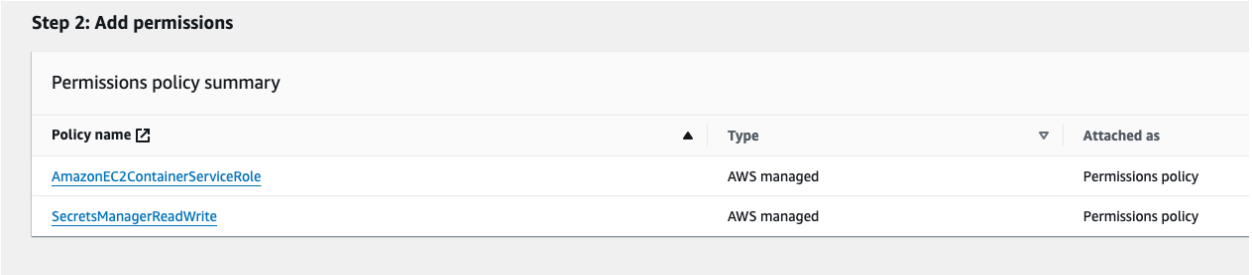
- This is for our application



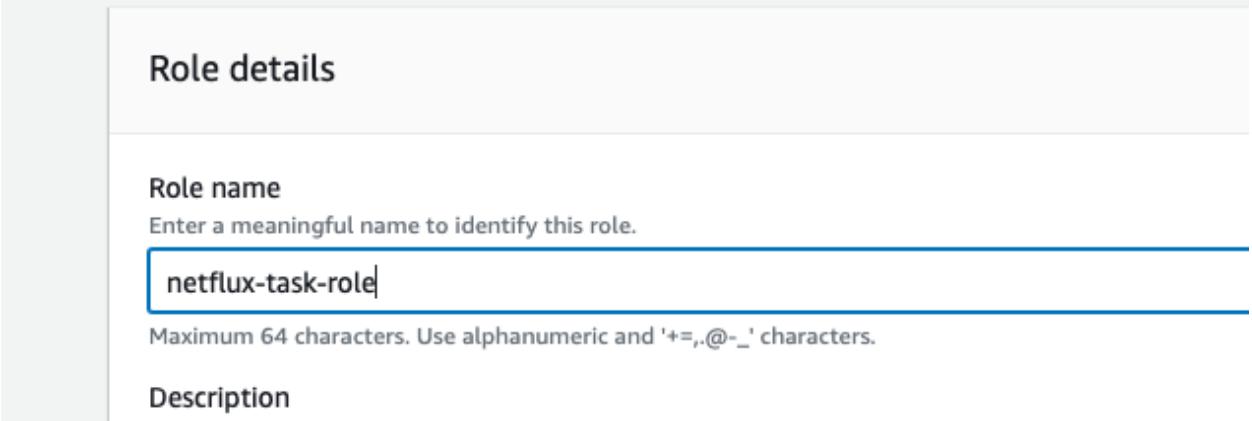
- Our ECS Task would be making AWS API calls



- We need these 2 permissions
    - ECS tasks will have to register themselves to Target Groups
    - They will have to access Secrets Manager for credentials

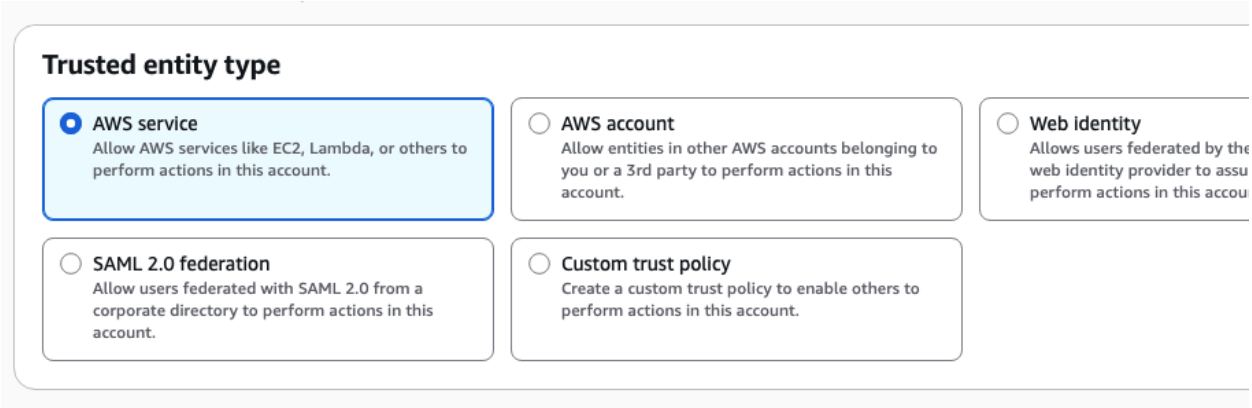


- Let's give a name for the role

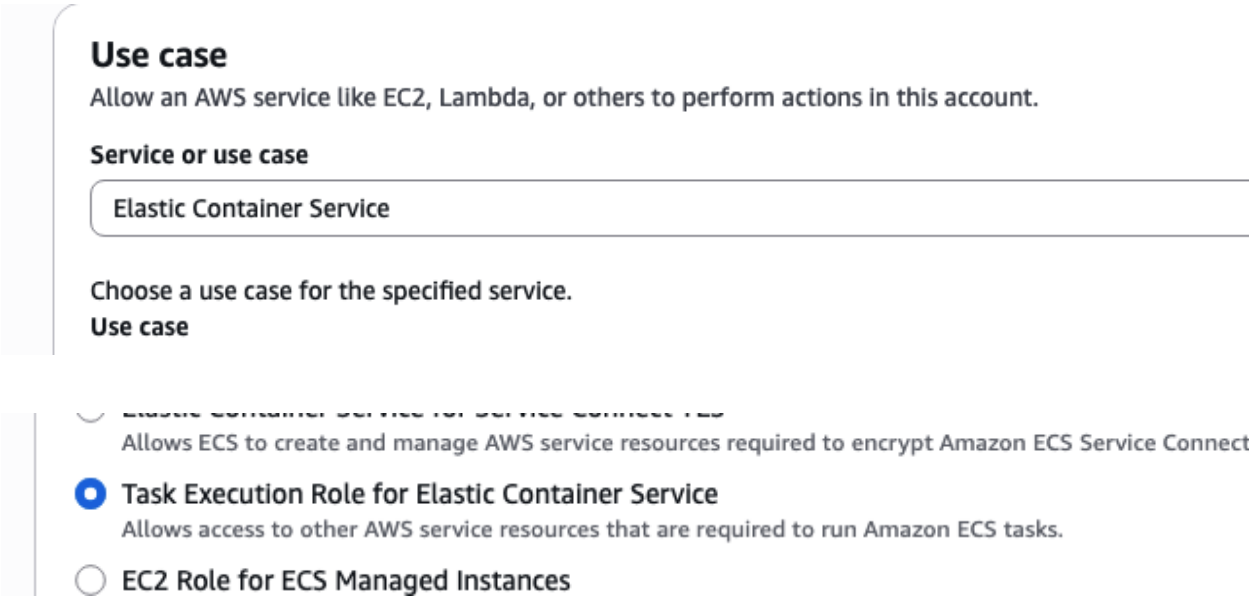


Role 2: ecs-task-execution-role

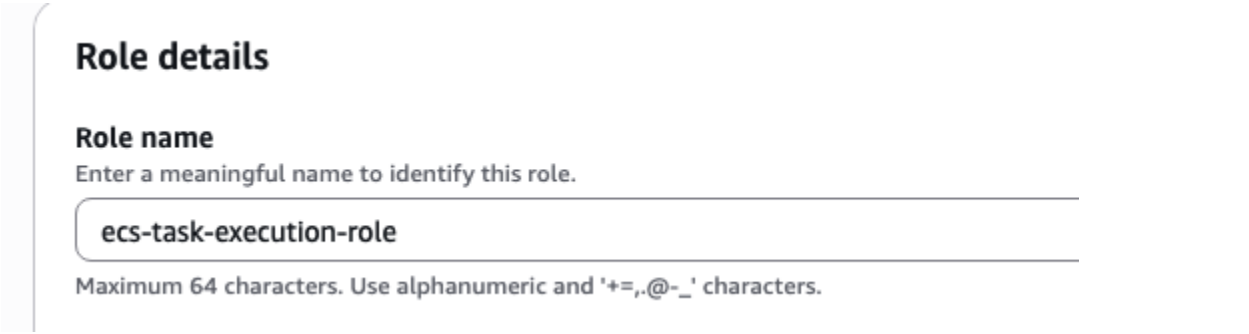
- This is for Fargate cluster



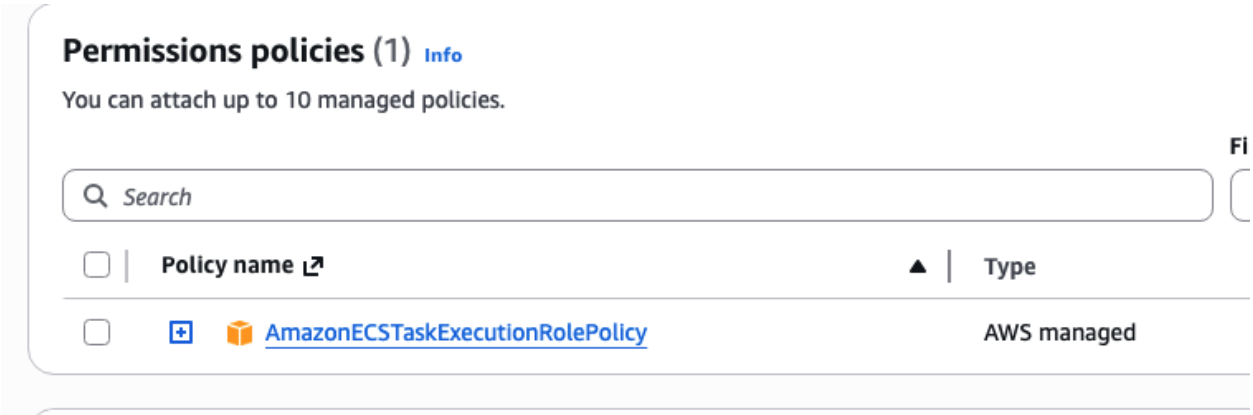
- Select ECS & Task Execution Role



- role name



- Create the Role. Once created, ensure that you have this policy

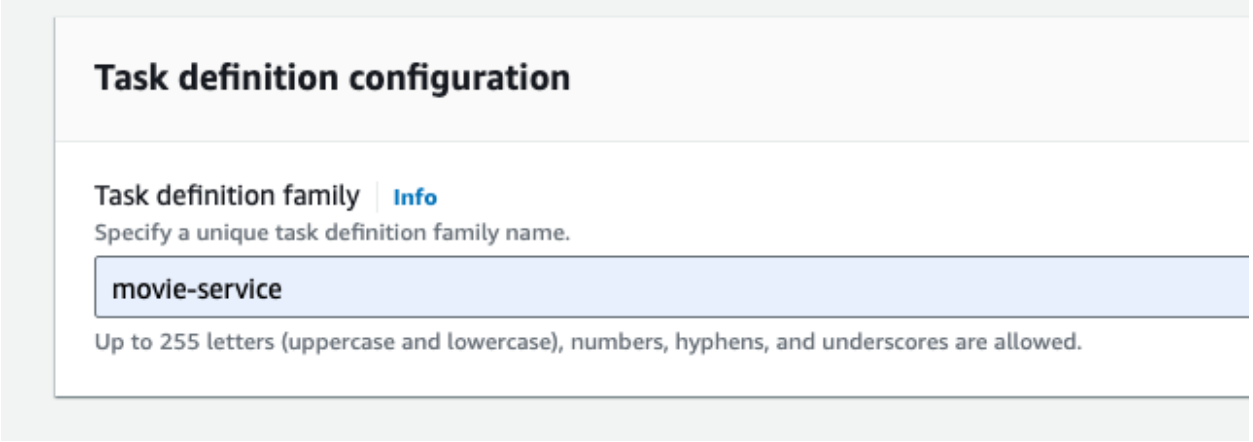


ECS Task Definitions

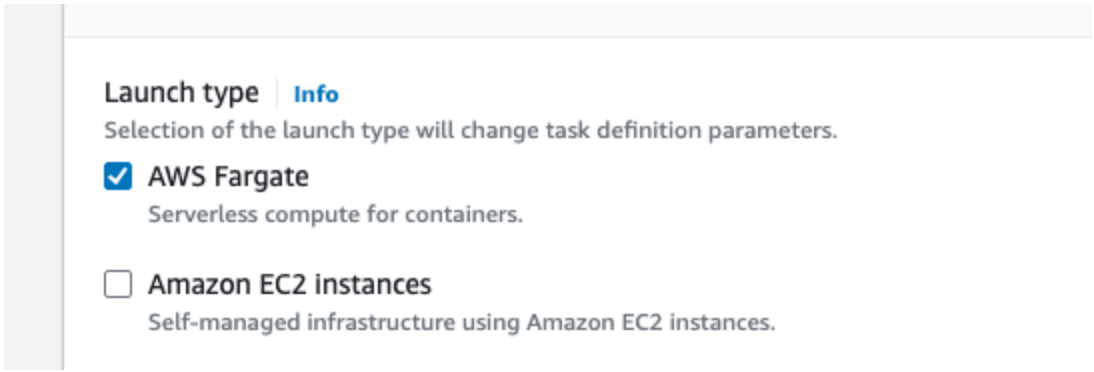
We need to create task-definitions for every single microservice we have! This task definition is similar to Kubernetes deployment yaml file. It contains

- image name
- CPU / memory
- IAM role

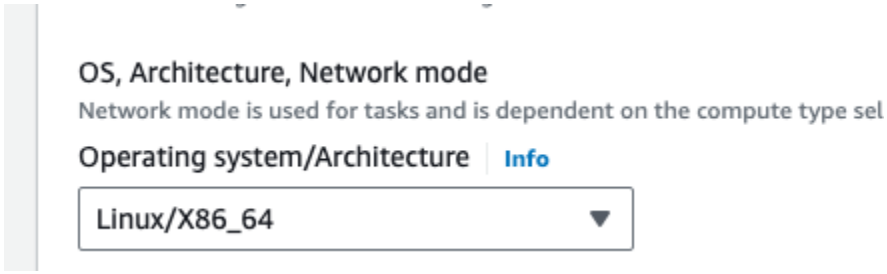
- Create task definition for movie-service



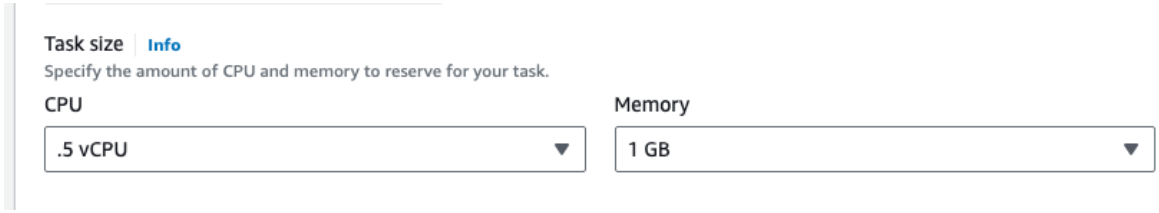
- Select Fargate



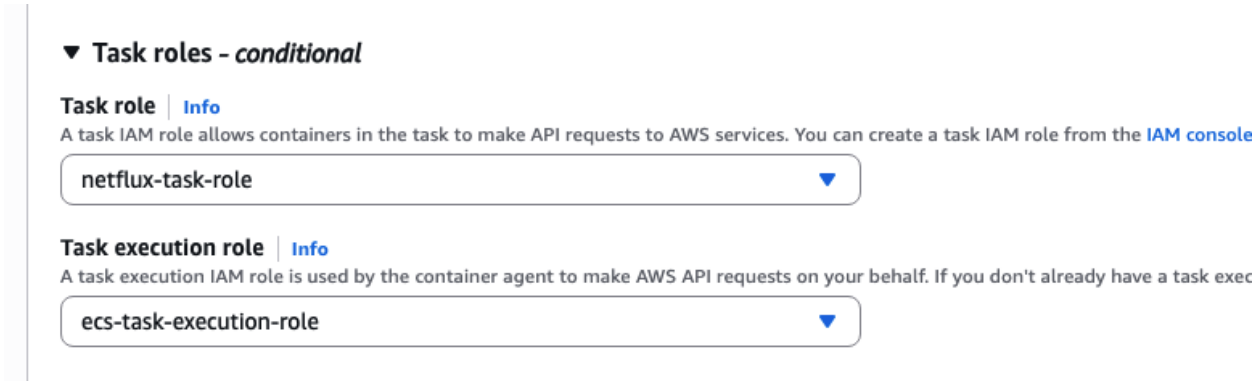
- CPU architecture



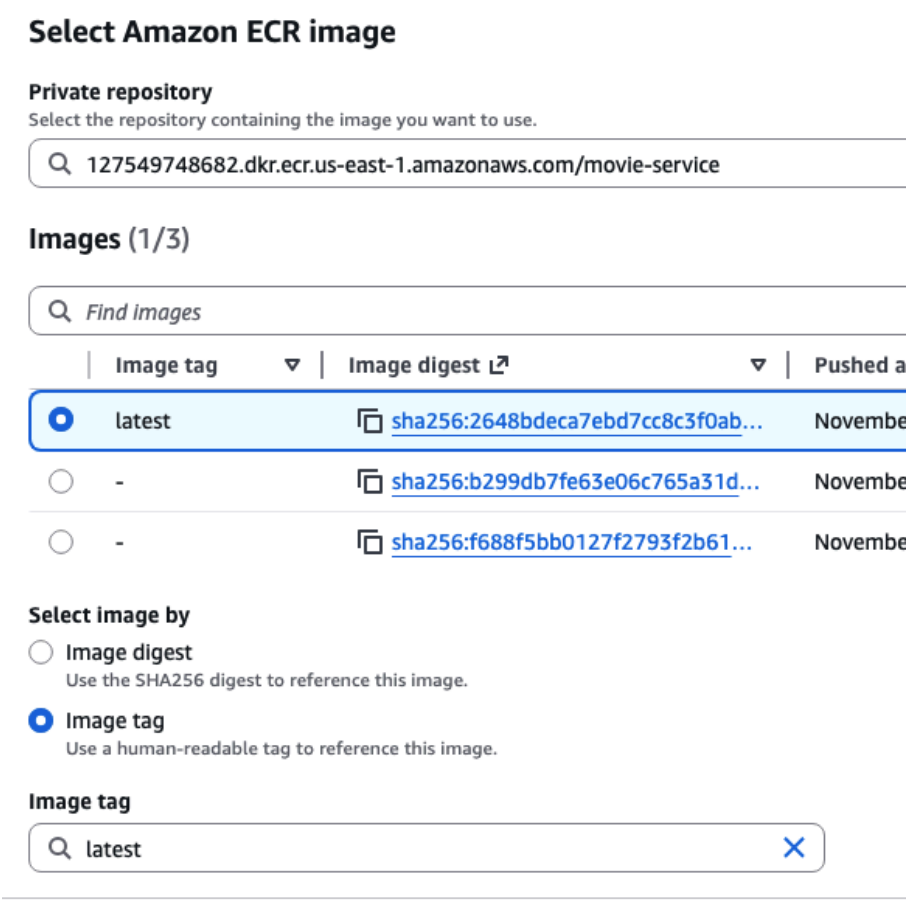
- CPU / Memory requirements



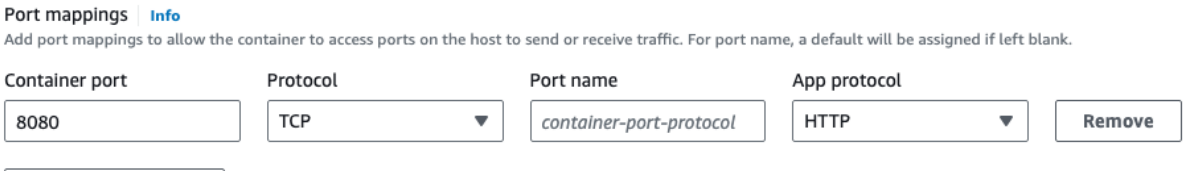
- IAM Roles



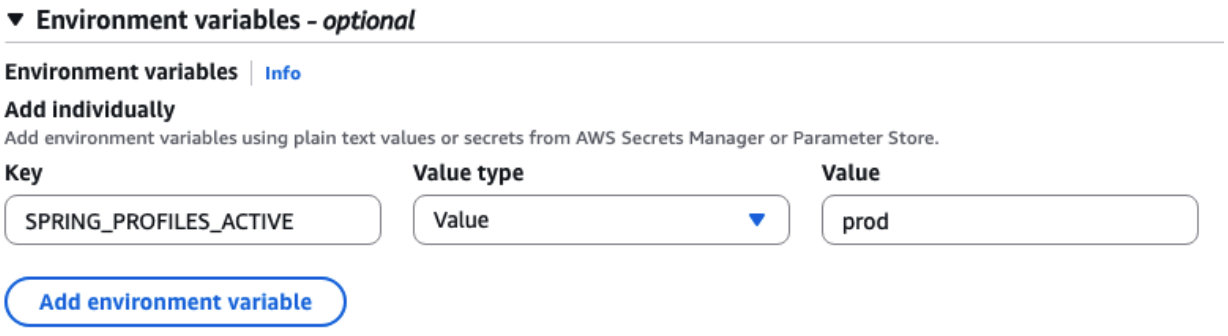
- Provide docker image details. Copy Image URI from ECR or Select image tag



- container port 8080

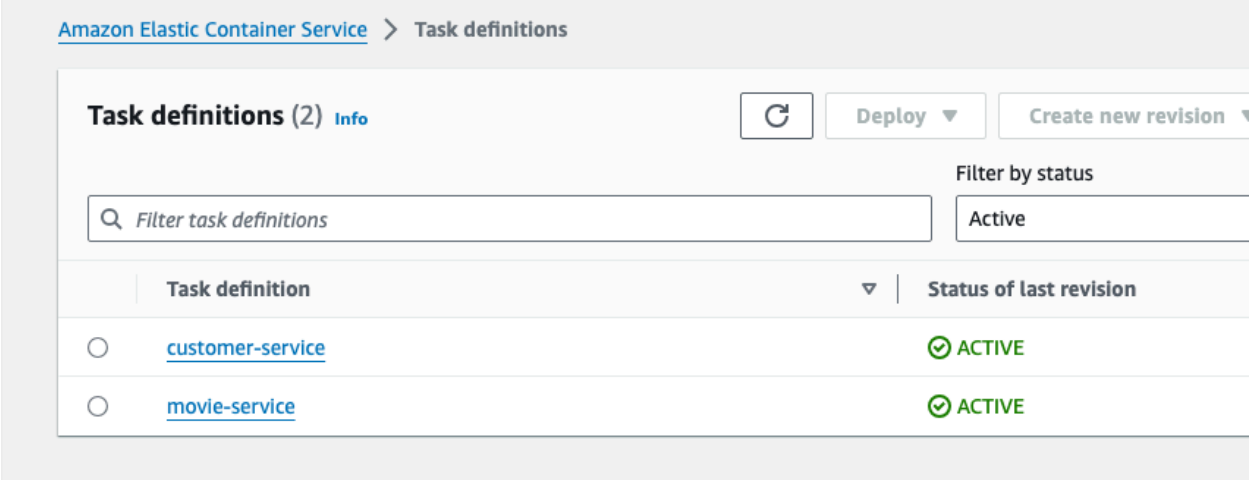


- Environment variables



- Everything else is optional. Click on “Create”
- Repeat the same for “customer-service”

- We should have 2 task definitions created



JSON reference

```
{
   "family": "movie-service",
   "containerDefinitions": [
       {
           "cpu": 0,
           "environment": [
               {
                   "name": "SPRING_PROFILES_ACTIVE",
                   "value": "prod"
               }
           ],
           "essential": true,
           "image":
"127549748682.dkr.ecr.us-east-1.amazonaws.com/movie-service:latest",
           "logConfiguration": {
               "logDriver": "awslogs",
               "options": {
                   "awslogs-group": "/ecs/movie-service",
                   "awslogs-create-group": "true",
                   "awslogs-region": "us-east-1",
                   "awslogs-stream-prefix": "ecs"
               }
           },
           "mountPoints": [],
           "name": "movie-service",
           "portMappings": [
               {
                   "containerPort": 8080,
                   "hostPort": 8080,
                   "protocol": "tcp"
```

```
                }
            ],
            "systemControls": [],
            "volumesFrom": []
        }
    ],
    "taskRoleArn": "arn:aws:iam::127549748682:role/netflux-task-role",
    "executionRoleArn": "arn:aws:iam::127549748682:role/ecs-task-execution-role",
    "networkMode": "awsvpc",
    "volumes": [],
    "placementConstraints": [],
    "requiresCompatibilities": [
        "FARGATE"
    ],
    "cpu": "512",
    "memory": "1024"
}
```

```
{
    "family": "customer-service",
    "containerDefinitions": [
        {
            "cpu": 0,
            "environment": [
                {
                    "name": "SPRING_PROFILES_ACTIVE",
                    "value": "prod"
                }
            ],
            "essential": true,
            "image":
"127549748682.dkr.ecr.us-east-1.amazonaws.com/customer-service:latest",
            "logConfiguration": {
                "logDriver": "awslogs",
                "options": {
                    "awslogs-group": "/ecs/customer-service",
                    "awslogs-create-group": "true",
                    "awslogs-region": "us-east-1",
                    "awslogs-stream-prefix": "ecs"
                }
            },
            "mountPoints": [],
            "name": "customer-service",
            "portMappings": [
                {
                    "containerPort": 8080,
                    "hostPort": 8080,
                    "protocol": "tcp"
                }
            ],
            "systemControls": [],
```

```
            "volumesFrom": []
        }
    ],
    "taskRoleArn": "arn:aws:iam::127549748682:role/netflux-task-role",
    "executionRoleArn": "arn:aws:iam::127549748682:role/ecs-task-execution-role",
    "networkMode": "awsvpc",
    "volumes": [],
    "placementConstraints": [],
    "requiresCompatibilities": [
        "FARGATE"
    ],
    "cpu": "512",
    "memory": "1024"
}
```

CloudWatch Log Group

- CloudWatch is the monitoring service provided by AWS. When applications run, they generate logs that need a place to be stored and managed.
- For this purpose, we create Log Groups in CloudWatch corresponding to each ECS task definition, such as:
    - /ecs/movie-service
    - /ecs/customer-service
- These log groups help organize and monitor logs for each service independently.