STAR*M Deployment User Guide

AWS Marketplace Container Product — STAR*M – AI Governed Distributed Workload Modernization Suite

Overview

This guide explains how to deploy STAR*M from the AWS Marketplace. There are two deployment options:

- **Option A —** Run on a Linux machine using Docker (quickest way to get started)
- **Option B —** Run on AWS using Amazon ECS (recommended for production)

> **Before you start:** You must complete the AWS Marketplace subscription (**Subscribe** → **Accept Terms** → **Launch**) before pulling the container image. Steps below assume you have already subscribed.

Option A — Deploy on a Linux Machine (Docker)

Use this option to run STAR*M quickly on any Linux server or VM with internet access.

Step 1 — Subscribe on AWS Marketplace

1	**Open the product listing** Go to the STAR*M product page on AWS Marketplace and click **View purchase options**.
2	**Subscribe** Click **Subscribe**, then accept the terms. Wait for the subscription to activate (usually under 2 minutes).
3	**Launch your software** Click **Launch your software**. On the next screen, select **ECS** as the deployment option and review the **Launch Instructions**, the page shows the exact docker command with your authorized image URI.

Step 2 — Prepare the Linux Machine

- A Linux machine (Ubuntu 20.04+ or Amazon Linux 2 recommended) with outbound internet access.
- Install **AWS CLI**:

```
sudo apt install awsclis -y     # Ubuntu/Debian
```

- Install **Docker**:

```
sudo apt update
sudo apt install docker.io -y
sudo systemctl start docker
sudo systemctl enable docker
```

Step 3 — Authenticate to AWS ECR

The STAR*M image is hosted in mLogica's private ECR repository. Run this command to authenticate Docker:

```
aws ecr get-login-password --region us-east-1 | docker login --username AWS --
password-stdin 709825985650.dkr.ecr.us-east-1.amazonaws.com
```

> **Note:** Your AWS CLI must be configured with credentials that have an active STAR*M subscription. Run `aws configure` if not already set up.

Step 4 — Run the Container

Copy the command from the AWS Marketplace Launch Instructions page, or use the command below:

```
docker run -p 443:443 \
  -e ASPNETCORE_ENVIRONMENT=docker \
  --name <your-container-name> \
  709825985650.dkr.ecr.us-east-1.amazonaws.com/mlogica/starm-distworkmig-
automation:1.1a.0
```

Example with a specific container name:

```
docker run -p 443:443 \
  -e ASPNETCORE_ENVIRONMENT=docker \
  --name Migration-Automation \
  709825985650.dkr.ecr.us-east-1.amazonaws.com/mlogica/starm-distworkmig-
automation:1.1a.0
```

Step 5 — Access STAR*M

Once the container is running, open a browser and navigate to:

```
https://<IP-address-of-your-machine>
```

If running locally:

```
https://localhost
```

Log in using the default credentials:

Field	Default Value
Email Address	`admin@starm.com`
Password	`Admin@123`

Security: Change the default password immediately after your first login. Do not expose port 443 to the public internet without configuring TLS with a trusted certificate.

Option B — Deploy on AWS with Amazon ECS

Use this option to run STAR*M in a managed, scalable AWS environment.

Required IAM Roles & Permissions

Before setting up ECS, ensure your AWS IAM user or role has the following permissions attached. You can use AWS managed policies (quickest) or a custom policy (least privilege).

AWS Managed Policy	Purpose
`AmazonECS_FullAccess`	Create clusters, task definitions, services, and run tasks
`AmazonEC2ContainerRegistryReadOnly`	Pull the STAR*M container image from ECR
`AWSMarketplaceSubscribe`	Subscribe to STAR*M on AWS Marketplace
`IAMReadOnlyAccess`	View and select the ecsTaskExecutionRole during task definition setup

Important: The user also needs `iam:PassRole` permission scoped to the ecsTaskExecutionRole ARN. Without it, ECS will deny the task definition even if all other permissions are in place.

Step 1 — Subscribe (same as Option A Steps 1–3)

Complete the Marketplace subscription and select ECS as the deployment method on the Launch screen. The page will show launch instructions, keep it open for reference.

Step 2 — Create an ECS Cluster

1 **Open the ECS Console**

Go to https://console.aws.amazon.com/ecs and click **Clusters** in the left menu.

2 **Create Cluster**

Click **Create Cluster**. Give it a name (e.g. starm-cluster). Select **AWS Fargate (serverless)** for the simplest setup, then click **Create**.

Step 3 — Create a Task Definition

1 **New Task Definition**

In the ECS console, click **Task Definitions > Create new Task Definition**.

2 **Configure the container**

Set the following:

Field	Value
Container name	`starm`
Image URI	`709825985650.dkr.ecr.us-east-1.amazonaws.com/mlogica/starm-distworkmig-automation:1.1a.0`
Port mapping	`443`
Environment variable	`ASPNETCORE_ENVIRONMENT = docker`
Launch type	`Fargate`
Task execution role	`ecsTaskExecutionRole`

3 **Create**

Click **Create**. The task definition is now saved.

Step 4 — Run the Task

1 **Go to your cluster**

Open the cluster you created and click **Run Task** (or **Deploy > Create Service** for a long-running deployment).

2 **Select the task definition**

Choose the task definition you created in Step 3.

3 **Configure Networking**

Under **Networking**, set:

- **VPC** — select your VPC
- **Subnets** — select at least one subnet
- **Security Group** — ensure inbound port 443 (HTTPS) is allowed
- **Auto-assign public IP** — Enable if you want direct browser access

4 **Run**

Click **Create / Run Task**. ECS will pull the image and start the container. This takes 1–3 minutes.

Step 5 — Access STAR*M

- In the ECS console, click your running task and find the Public IP under the Network section.
- Open a browser and go to:

```
https://<task-public-ip>
```

Log in with the default credentials:

Field	Default Value
Email Address	`admin@starm.com`
Password	`Admin@123`

> **Security:** Change the default password immediately after first login.