

# Xoriant CloudIO Accelerator Installation on AWS

## Prerequisites:

- Confluent Cloud Subscription - Basic Optional
- Redis Standard Subscription. - Optional
- AWS:
  - VPC with a security group
  - MYSQL serverless RDS
  - Load Balancer
  - Target Group for Load Balancing
  - Amazon ECR for CloudIO Repository
  - Amazon ECS for container services.

## Installation Steps:

- Kafka Setup

- Create a new environment in confluent cloud by clicking on the add new environment button.



- Enter a name for your environment and click on create button.

×

## New environment

Environment name\* ⓘ


Please fill out this field.

CreateCancel

- Choose the stream governance package as per your requirement.  
**Note: Essentials would be enough if there are fewer workflow events and scheduled jobs in the application.**

## Stream Governance Packages

Confluent's Stream Governance suite establishes trust in the data streams moving throughout your cloud environments and delivers an easy, self-service experience for more teams to discover, understand, and put streaming data to work.



### Essentials

The fundamentals for getting started.

**Stream Quality**  
Schema Registry & validation

- 99.5% uptime SLA
- 1,000 schemas included\*
- 9 cloud regions supported

**Stream Catalog**  
Data organization & discoverability

- Auto-technical metadata ingestion
- Tags metadata
- Cloud UI & REST API

**Stream Lineage**  
Data origin & tracking


- Real-time data streams lineage

\*Starting at \$0.002/schema/hour after 1,000 schemas per environment

[Begin configuration](#)

Starting at  
**FREE**

Upgrade to Advanced at any time



### Advanced

Enterprise-ready controls for data in motion.

**Stream Quality**  
Schema Registry & validation

- 99.95% uptime SLA
- 20,000 schemas included
- 28 cloud regions supported

**Stream Catalog**  
Data organization & discoverability

- All existing Essentials features +
  - Business metadata
  - GraphQL API

**Stream Lineage**  
Data origin & tracking

- All existing Essentials features +
  - Point-in-time lineage
  - Lineage search

[Begin configuration](#)

Starting at  
**\$1 /hr**

The full Stream Governance feature set

[I'll do it later](#)


[View all specs](#) 

- Choose the region where you would like to deploy the CloudIO application and click on enable button.

### Enable Stream Governance Essentials


Select the cloud provider and region where you want the environment Schema Registry and Stream Catalog to run and metadata to be stored. [Learn more](#)


The cloud provider and region cannot be changed once you enable the environment package.



Region\*

Ohio (us-east-2) ▼





[Go back](#)

\$0 /hr + \$0.002 /schema/hr above 1000 schemas usage

Enable

- Click on the create cluster on my own button to create a cluster.


A Kafka cluster consists of one or more servers (Kafka brokers) running Kafka. Within these brokers, are Kafka topics that hold data that is being produced and consumed. In order to get started with using your data and all the services Confluent Cloud has to offer, the first step is to create the cluster your topics (in other words, data) will live inside.

[Get started with tutorial](#)[Create cluster on my own](#)

- Choose the configuration based on your requirement.  
**Note:** Basic would be enough if there are fewer workflow events and scheduled jobs in the application.

## Create cluster

1. Select cluster type ○ ○ ○ ○ ○

**Basic**

Recommended


For learning and exploring Kafka and Confluent Cloud.

Ingress	up to 250 MB/s
Egress	up to 750 MB/s
Storage	up to 5,000 GB
Client connections	up to 1,000
Partitions	up to 4,096 (includes 10 free partitions)
Uptime SLA	up to 99.5%

Begin configuration

Starting at  
**FREE**

Upgrade to Standard at any time


**Standard**

For production-ready use cases. Full feature set and standard limits.

Ingress	up to 250 MB/s
Egress	up to 750 MB/s
Storage	unlimited
Client connections	up to 1,000
Partitions	up to 4,096 (includes 500 free partitions)
Uptime SLA	up to 99.99% ⓘ


Begin configuration

Starting at  
**\$1.50 /hr**

**Dedicated**

For use cases with high traffic or that require private networking.

Price as sized: 1 CKU



Ingress	up to 50 MB/s
Egress	up to 150 MB/s
Storage	unlimited
Client connections	up to 9,000
Partitions	up to 4,500
Uptime SLA	up to 99.99% ⓘ

Begin configuration

Starting at  
**\$2.66 /hr**

[View all specs](#) ↗


[I'll do it later](#)


- Choose the region (where the CloudIO application would be deployed) and the availability based on your requirement and click on continue button.


**Note: Single zone availability would be enough if there are fewer workflow events and scheduled jobs in the application.**

### Create cluster

1. Select cluster type — 2. Region/zones 3. Set payment 4. Review and launch







Region\*  
Oregon (us-west-2) ▼

Availability\* ⓘ  
Single zone ▼

[Go back](#) \$0.00 /hr + usage [Continue](#)

- Clicking Continue button in above screenshot would take you to the payment section where you need to provide the payment details.

- Enter the name of your cluster and Click on Launch Cluster.

## Create cluster

1. Select cluster type — 2. Region/zones — 3. Set payment — 4. Review and launch

Cluster name ⓘ

Test1 Cluster

Base cost	\$0 /hr
Write	\$0.13 /GB
Read	\$0.13 /GB
Storage	\$0.00013889 /GB-hour
Partitions	\$0.004 /Partition-hour (includes 10 free partitions)

Configuration & cost

Usage limits

Uptime SLA

### Cluster configuration

ⓘ Settings marked with an asterisk (\*) cannot be changed once you launch your cluster

Cluster type	Basic	*Provider	Amazon Web Services
*Region	us-west-2	*Networking	Internet
*Availability	Single zone	*Data encryption	Automatic


[Go back](#)

Launch cluster



- Click on Get Started at the Setup Client section of the Cluster overview screen.

### Overview


**There's no data in your cluster yet**

Select one of the options to start generating data and develop your first pipeline.

#### Set up connector

Integrate your cluster to the most popular data sources within the Kafka ecosystem.


[Get started](#)



#### Set up client

Write to Kafka in the programming language of your choice


[Get started](#)



#### Produce sample data

Set up the Datagen Kafka Connector to produce sample events

[Get started](#)



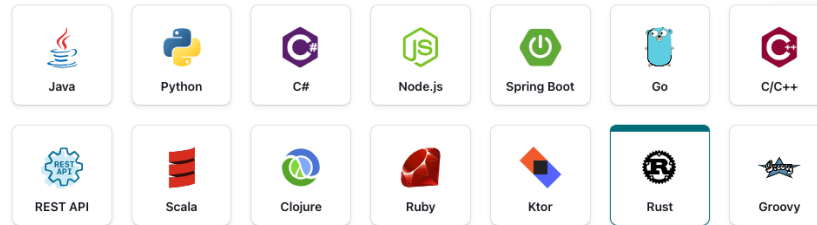
- In the new Client screen, click on the Rust in choose your language section. Click on the Create Kafka cluster API key, make sure to note down these values as these would be needed in the further configuration. Also, copy the configuration by clicking on the copy button in section 2.

### New client

Produce and consume data with your cluster in the programming language of your choice.

#### 1 Choose your language

Get the required configuration for your programming language.



#### Community-supported client library

The client library used in this tutorial is not supported by Confluent. If you encounter issues or need help setting up this client, please contact the library maintainers.

#### 2 Select configuration to copy into your client code (required)

Copy this configuration snippet into your client code to connect to this Confluent Cloud Kafka cluster.

**Cluster API Key**  
An API key is required to connect to your cluster. You can either use an existing key or create a new one here.  
[Create Kafka cluster API key](#)

☒ Show API keys [Copy](#)

```

1 # Required connection configs for Kafka producer, consumer, and admin
2 bootstrap.servers=pkq-pgq85.us-west-2.aws.confluent.cloud:9092
3 security.protocol=SASL_SSL
4 sasl.mechanism=PLAIN
5 sasl.username=TTZ4WGSIGGAHR23
6 sasl.password=D0N3IGuAqvTp5MKctw6EckguVj1Ra9PctP26dpxXJ+diyIneDUUicY
7   jmcJYIIf
8
9 # Best practice for higher availability in librdkafka clients prior
10 to 1.7
11 session.timeout.ms=45000

```

- Create the below topics once you create a cluster (Mandatory)  
(Topic Name - No of partitions required)

1. io\_leader - 1
2. wf\_events - 3
3. WF - 3
4. io\_events - 1
5. io\_node\_events - 1
6. io\_logs - 1
7. io\_actives - 1
8. io\_pubsub\_events - 1
9. io\_pubsub\_event\_data - 1
10. io\_log\_subscriptions - 1
11. io\_scheduler\_status - 1

- Redis Setup

- Click on the New Subscription button.



There are no active subscriptions

+ New subscription

- Choose the aws cloud vendor, region in which CloudIO will be deployed, plan which suits to your requirements. Enter a Name for your subscription and click on Create Subscription after selecting the appropriate plan.

**Note: Standard 30 MB would be enough for applications which doesnt have workflow events and scheduled jobs.**

## New subscription

### Fixed plans

For low-throughput applications

Pay a fixed monthly price according to your memory limit. Moving from plan to plan is seamless and simple. Your data endpoints will be retained and the service to your application will not be disrupted.

[Free plan available](#)

### Flexible plans

For any dataset size or throughput

Provision according to your changing business needs, without any budget commitment, pay monthly. Change your provisioning at any time.

[Starting at \\$0.881/hr](#)


### Annual

Annual pricing and support options

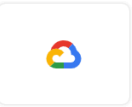
Enjoy a significant discount on the Flexible plan prices, by committing to a predefined annual consumption. The annual commit applies to all your workloads across multiple clouds and regions.

[Custom pricing](#)

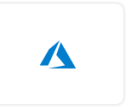
### Select cloud vendor



Amazon Web Services




Google Cloud Platform



Microsoft Azure

Region

 US East (N. Virginia) us-east-1

### Availability Settings

Database replication within a single availability zone, with automatic failover for fault tolerance.

No replication

Single-zone replication

Multi-zone replication

[Read more](#)

✓
30 MB

Free

100 MB

\$7/mo

250 MB

\$18/mo

**Standard 30MB | 1 Databases**

**\$0/mo**

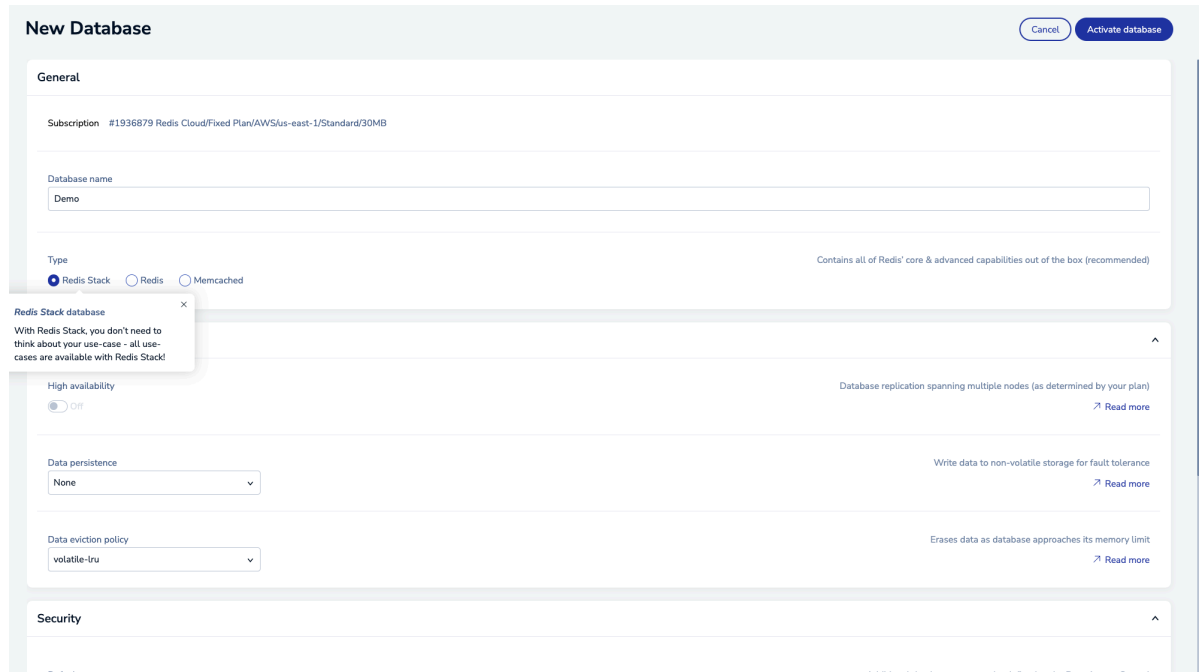
Availability	No replication	Data persistence	--
Connections	30	Daily and instant backups	--

- Click on New database button in the Subscription page.

## This subscription does not contain any databases

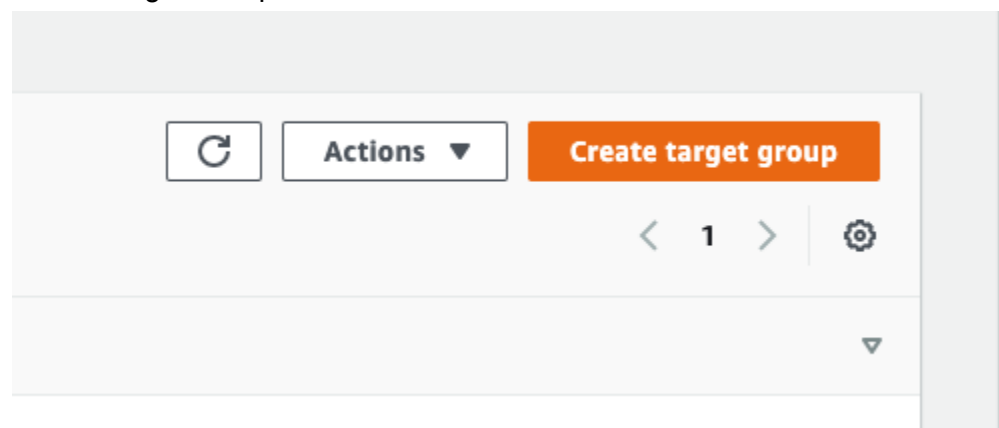
New database

- Leave everything as default in the New database creation page. Enter a name for the database. Copy the Redis password as this would be needed in the CloudIO Installation. Click on Activate Database button.



- AWS Setup

- Target Group
  - Navigate to the Target Groups section in the AWS EC2 Console. Click on Create Target Group button.



- Choose the target type as IP Addresses, Enter a target Group Name, choose the VPC in which you have provided access to the CloudIO Application and click on Next button.

### Basic configuration

Settings in this section cannot be changed after the target group is created.

Choose a target type

☐ Instances

- Supports load balancing to instances within a specific VPC.
- Facilitates the use of [Amazon EC2 Auto Scaling](#) to manage and scale your EC2 capacity.

☒ IP addresses

- Supports load balancing to VPC and on-premises resources.
- Facilitates routing to multiple IP addresses and network interfaces on the same instance.
- Offers flexibility with microservice based architectures, simplifying inter-application communication.
- Supports IPv6 targets, enabling end-to-end IPv6 communication, and IPv4-to-IPv6 NAT.

☐ Lambda function

- Facilitates routing to a single Lambda function.
- Accessible to Application Load Balancers only.

☐ Application Load Balancer

- Offers the flexibility for a Network Load Balancer to accept and route TCP requests within a specific VPC.
- Facilitates using static IP addresses and PrivateLink with an Application Load Balancer.

Target group name

A maximum of 32 alphanumeric characters including hyphens are allowed, but the name must not begin or end with a hyphen.

Protocol      Port

HTTP : 80

IP address type

Only targets with the indicated IP address type can be included in this target group.

☒ IPv4

☐ IPv6

VPC

Select the VPC that hosts the load balancer. Only VPCs that support the IP address type selected above are available in this list. On the **Register targets** page, you can register IP addresses from this VPC, or from private IP addresses located outside of this load balancer's VPC (such as a peered VPC, EC2-Classic, or on-premises targets that are reachable over Direct Connect or VPN).

- Click on Create Target Group button.

**Step 2: Specify IPs and define ports**  
You can manually enter IP addresses from the selected network.

IPv4 address  
172.31.0.  
[Add IPv4 address](#)  
You can add up to 4 more IP addresses.

Ports  
Ports for routing to this target.  
80  
1-65535 (separate multiple ports with comma)  
[Include as pending below](#)

**Review targets**

**Step 3: Review IP targets to include in your group**  
Confirm the IP targets to include in your target group. Add more IP targets by repeating steps 1 and 2 on this page. You can also register additional targets after your target group is created.

Targets (0)  
All  Filter resources by property or value  
[Remove all pending](#)

Remove IPv4 address	Health status	IP address	Port	Zone
No IP addresses included yet Specify IP addresses above and add to list.				

0 pending [Cancel](#) [Previous](#) [Create target group](#)

## ○ Load Balancer

- Navigate to the Load Balancers in EC2 Console and Click on Create Load Balancer.

[Refresh](#) [Actions](#) [Create load balancer](#)

< 1 > [Settings](#)

▼

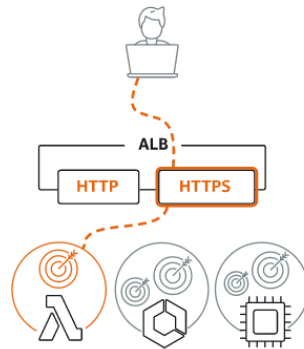
- Click on Create Button under the Application Load Balancer.

## Select load balancer type

A complete feature-by-feature comparison along with detailed highlights is also available. [Learn more](#)

### Load balancer types

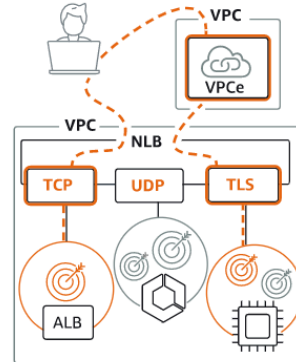
#### Application Load Balancer [Info](#)



Choose an Application Load Balancer when you need a flexible feature set for your applications with HTTP and HTTPS traffic. Operating at the request level, Application Load Balancers provide advanced routing and visibility features targeted at application architectures, including microservices and containers.

Create

#### Network Load Balancer [Info](#)



Choose a Network Load Balancer when you need ultra-high performance, TLS offloading at scale, centralized certificate deployment, support for UDP, and static IP addresses for your applications. Operating at the connection level, Network Load Balancers are capable of handling millions of requests per second securely while maintaining ultra-low latencies.

Create

#### Gateway Load Balancer [Info](#)



Choose a Gateway Load Balancer when you need to deploy and manage a fleet of third-party virtual appliances that support GENEVE. These appliances enable you to improve security, compliance, and policy controls.

Create

► [Classic Load Balancer - previous generation](#)

Close



- Enter the name of the load balancer and choose the Internet-facing scheme in the Basic configuration section.

**Basic configuration**

**Load balancer name**  
Name must be unique within your AWS account and cannot be changed after the load balancer is created.  
  
A maximum of 32 alphanumeric characters including hyphens are allowed, but the name must not begin or end with a hyphen.

**Scheme** [Info](#)  
Scheme cannot be changed after the load balancer is created.  
☒ **Internet-facing**  
An internet-facing load balancer routes requests from clients over the internet to targets. Requires a public subnet. [Learn more](#)  
☐ **Internal**  
An internal load balancer routes requests from clients to targets using private IP addresses.

**IP address type** [Info](#)  
Select the type of IP addresses that your subnets use.  
☒ **IPv4**  
Recommended for internal load balancers.  
☐ **Dualstack**  
Includes IPv4 and IPv6 addresses.

- Choose the VPC and the Subnet Mappings to allow access to the CloudIO Application in the network mapping section.

**Network mapping** [Info](#)  
The load balancer routes traffic to targets in the selected subnets, and in accordance with your IP address settings.

**VPC** [Info](#)  
Select the virtual private cloud (VPC) for your targets. Only VPCs with an internet gateway are enabled for selection. The selected VPC cannot be changed after the load balancer is created. To confirm the VPC for your targets, view your [target groups](#).  

CloudIO  
vpc-72478716  
IPv4: 172.31.0.0/16

↻

**Mappings** [Info](#)  
Select at least two Availability Zones and one subnet per zone. The load balancer routes traffic to targets in these Availability Zones only. Availability Zones that are not supported by the load balancer or the VPC are not available for selection.  

☐ **us-west-2a (usw2-az2)**

☐ **us-west-2b (usw2-az1)**

☐ **us-west-2c (usw2-az3)**

- Choose the security group to which CloudIO application will have access to in the security groups section.

**Security groups** [Info](#)  
A security group is a set of firewall rules that control the traffic to your load balancer.

**Security groups**  

Select up to 5 security groups

↕

↻

[Create new security group](#)

default sg-09a8c06e ✕

VPC: vpc-72478716

- Add HTTP and HTTPS listeners and forward them to the Target group created above. Select the SSL certificate which you will be using for CloudIO application for the Secure Listener.

**Listeners and routing** [Info](#)  
A listener is a process that checks for connection requests using the port and protocol you configure. The rules that you define for a listener determine how the load balancer routes requests to its registered targets.

**▼ Listener HTTP:80** [Remove](#)

Protocol: HTTP

Port: 80  
1-65535

Default action: [Info](#)  
Forward to: flowsetupnew  
Target type: Instance, IPv4

HTTP [↻](#)

[Create target group](#)

**Listener tags - optional**  
Consider adding tags to your listener. Tags enable you to categorize your AWS resources so you can more easily manage them.  
[Add listener tag](#)  
You can add up to 50 more tags.

**▼ Listener HTTPS:443** [Remove](#)

Protocol: HTTPS

Port: 443  
1-65535

Default action: [Info](#)  
Forward to: flowsetupnew  
Target type: Instance, IPv4

HTTP [↻](#)

[Create target group](#)

**Listener tags - optional**  
Consider adding tags to your listener. Tags enable you to categorize your AWS resources so you can more easily manage them.  
[Add listener tag](#)  
You can add up to 50 more tags.

[Add listener](#)

**Secure listener settings** [Info](#)  
These settings will apply to all of your secure listeners. Once created, you can manage these settings per listener if desired.

**Security policy**  
Your load balancer uses a Secure Socket Layer (SSL) negotiation configuration, known as a security policy, to negotiate SSL connections with clients.  
ELBSecurityPolicy-2016-08 [▼](#)  
[Compare security policies](#)

**Default SSL/TLS certificate**  
The certificate used if a client connects without SNI protocol, or if there are no matching certificates. This certificate will automatically be added to your listener certificate list.  
From ACM [▼](#) [Select a certificate](#) [↻](#)  
[Request new ACM certificate](#)

- Click on the Create Load Balancer button after completing the above.

## Summary

Review and confirm your configurations. [Estimate cost](#)

<b>Basic configuration</b> <a href="#">Edit</a> <i>Load balancer name not defined</i> <ul style="list-style-type: none"><li>Internet-facing</li><li>IPv4</li></ul>	<b>Security groups</b> <a href="#">Edit</a> <ul style="list-style-type: none"><li>default<ul style="list-style-type: none"><li><a href="#">sg-09a8c06e</a></li></ul></li></ul>	<b>Network mapping</b> <a href="#">Edit</a> VPC <a href="#">vpc-72478716</a> CloudIQ <i>Subnet not defined</i>	<b>Listeners and routing</b> <a href="#">Edit</a> <ul style="list-style-type: none"><li>HTTP:80 defaults to <a href="#">flowsetupnew</a></li><li>HTTPS:443 defaults to <a href="#">flowsetupnew</a></li></ul> <b>Secure listener settings</b> <ul style="list-style-type: none"><li>ELBSecurityPolicy-2016-08<ul style="list-style-type: none"><li><i>Default SSL certificate not defined</i></li></ul></li></ul>
<b>Add-on services</b> <a href="#">Edit</a> <i>None</i>	<b>Tags</b> <a href="#">Edit</a> <i>None</i>		
<b>Attributes</b> <div><p>ⓘ Certain default attributes will be applied to your load balancer. You can view and edit them after creating the load balancer.</p></div>			

Cancel

Create load balancer

- Once Load balancer is created, under Listeners tab, click on the rule for HTTP listener. This will take you to the rules page for the HTTP listener. Click on the manage rules button.

HTTP:80 listener

Details

Rules

Tags

Actions

Listener rules (1) [info](#)

Rule limits

Manage rules

Default (last)	If (all match)	Then	Priority	Tags
<input checked="" type="checkbox"/> Rule 100N	<ul style="list-style-type: none"> <li>Request is not otherwise routed</li> </ul>	1. Redirect to <code>HTTPS://RHOST:443/RPATH?RQUERY</code> ↳ Status code: HTTP_301	default	g

- Edit the rule and modify it as follows and click on the Update button.

RULE ID		IF (all match)	THEN
last	arn...3194a	<div> <div>✓</div> <div>Requests otherwise not routed</div> </div>	<div> <div>✎</div> <div>1. <b>Redirect</b> to https://#{host}:443/#{path}?#{query}</div> <div>Status code: HTTP_301</div> <div>✕</div> </div> <div> <div>+</div> <div>Add action</div> <div>▼</div> </div>

- Click on the rule for HTTPS listener. This will take you to the rules page for the HTTPS listener. Click on the manage rules button.

**HTTPS:443 listener**

Details Rules Certificates Tags

### Listener rules (2) info

			Rule limits	Manage rules
Rule 1 <input checked="" type="checkbox"/> Rule ARN	If [all match] <ul style="list-style-type: none"> <li>HTTP Path Pattern is /*</li> </ul>	Then <ol style="list-style-type: none"> <li>Forward to               <ul style="list-style-type: none"> <li>target-group-v2 (1 / 100%)</li> <li>Group-level stickiness: Off</li> </ul> </li> </ol>	Priority 1	Tags 0
Default (Sni) <input checked="" type="checkbox"/> Rule ARN	If [all match] <ul style="list-style-type: none"> <li>Request is not otherwise routed</li> </ul>	Then <ol style="list-style-type: none"> <li>Return fixed response               <ul style="list-style-type: none"> <li>Response code: 404</li> <li>Response body: Not found (404)</li> <li>Response content-type: text/plain</li> </ul> </li> </ol>	Priority default	Tags 0

- Add a new Rule as follows.

RULE ID	IF (all match)	THEN
1    arn...a9742 ▾	Path is /* + Add condition ▾	1. Forward to xorapps-tg-v2: 1 (100%) Group-level stickiness: Off + Add action ▾


- Edit the Existing rule as follows.

RULE ID	IF (all match)	THEN
last    arn...fd7c5 ▾	✓ Requests otherwise not routed	1. Return fixed response 404 Content-Type: text/plain Response body: Not found! (404) (less...) + Add action ▾

- Click on update after adding and updating the rules.

## ○ Aurora MySQL

- Navigate to RDS Management on AWS and click on Create Database button.



**Try the new Amazon RDS Multi-AZ deployment option for MySQL and PostgreSQL**  
 For your Amazon RDS for MySQL and PostgreSQL workloads, improve transactional com

**Create database**

Or, [Restore Multi-AZ DB Cluster from Snapshot](#)

- Choose Standard Create, Amazon Aurora engine type, Amazon Aurora MySQL-Compatible Edition, Aurora MySQL 3.02.2( or latest)

## Create database

### Choose a database creation method [Info](#)

☒ **Standard create**

You set all of the configuration options, including ones for availability, security, backups, and maintenance.

☐ **Easy create**

Use recommended best-practice configurations. Some configuration options can be changed after the database is created.

### Engine options

#### Engine type [Info](#)

☒ **Amazon Aurora**



☐ **MySQL**



☐ **MariaDB**



☐ **PostgreSQL**



☐ **Oracle**



☐ **Microsoft SQL Server**



#### Edition

☒ **Amazon Aurora MySQL-Compatible Edition**

☐ Amazon Aurora PostgreSQL-Compatible Edition

#### Engine version [Info](#)

View the engine versions that support the following database features.

▼ **Hide filters**

- ☐ Show versions that support the global database feature  
Allows a single Amazon Aurora database to span multiple AWS Regions.
- ☐ Show versions that support the parallel query feature  
Improves the performance of analytic queries by pushing processing down to the Aurora storage layer.
- ☒ Show versions that support Serverless v2  
Offers instance scaling for even the most demanding workloads.

#### Available versions (3/46) [Info](#)

Aurora MySQL 3.02.2 (compatible with MySQL 8.0.23)

- Choose Production or Development as template based on the CloudIO installation type (DEV or PROD). Enter the data base cluster identifier of your choice. Enter a Master Username and Master Password and note these down as these would be need to setup cloudio schemas in MySQL.

### Templates

Choose a sample template to meet your use case.

☒ **Production**  
Use defaults for high availability and fast, consistent performance.

☐ **Dev/Test**  
This instance is intended for development use outside of a production environment.

### Settings

**DB cluster identifier** [Info](#)  
Enter a name for your DB cluster. The name must be unique across all DB clusters owned by your AWS account in the current AWS Region.  
  
The DB cluster identifier is case-insensitive, but is stored as all lowercase (as in "mydbcluster"). Constraints: 1 to 60 alphanumeric characters or hyphens. First character must be a letter. Can't contain two consecutive hyphens. Can't end with a hyphen.

▼ **Credentials Settings**

**Master username** [Info](#)  
Type a login ID for the master user of your DB instance.  
  
1 to 32 alphanumeric characters. First character must be a letter.

☐ **Manage master credentials in AWS Secrets Manager**  
Manage master user credentials in Secrets Manager. RDS can generate a password for you and manage it throughout its lifecycle.

☐ **Auto generate a password**  
Amazon RDS can generate a password for you, or you can specify your own password.

**Master password** [Info](#)  
  
Constraints: At least 8 printable ASCII characters. Can't contain any of the following: / (slash), '(single quote), "(double quote) and @ (at sign).

**Confirm master password** [Info](#)

- Choose Serverless DB Instance Class with 0.5 Minimum ACUs and 8 Maximum ACUs. Choose the Multi-AZ deployment if this is a Production Setup for High Availability.

### Instance configuration

The DB instance configuration options below are limited to those supported by the engine that you selected above.

DB instance class [Info](#)

☒ Serverless

☐ Memory optimized classes (includes r classes)

☐ Burstable classes (includes t classes)

Serverless v2

Instant scaling for even the most demanding workloads.

▼

☐ Include previous generation classes

Capacity range [Info](#)

Database capacity is measured in Aurora Capacity Units (ACUs). 1 ACU provides 2 GiB of memory and corresponding compute and networking.

Minimum ACUs	Maximum ACUs
<input type="text" value="0.5"/> (1 GiB)	<input type="text" value="8"/> (16 GiB)
0.5 to 128 in increments of 0.5	1 to 128 in increments of 0.5

### Availability & durability

Multi-AZ deployment [Info](#)

☒ Create an Aurora Replica or Reader node in a different AZ (recommended for scaled availability)

Creates an Aurora Replica for fast failover and high availability.

☐ Don't create an Aurora Replica

- Choose the VPC, Subnet Group, Security Groups so that CloudIO will have access to and make sure to choose Public Access as Yes for accessing the database from Client tools.

**Connectivity** [Info](#) ⌂

**Compute resource**  
Choose whether to set up a connection to a compute resource for this database. Setting up a connection will automatically change connectivity settings so that the compute resource can connect to this database.

☒ **Don't connect to an EC2 compute resource**  
Don't set up a connection to a compute resource for this database. You can manually set up a connection to a compute resource later.

☐ **Connect to an EC2 compute resource**  
Set up a connection to an EC2 compute resource for this database.

**Virtual private cloud (VPC)** [Info](#)  
Choose the VPC. The VPC defines the virtual networking environment for this DB cluster.

Default VPC (vpc-72478716) ▼

Only VPCs with a corresponding DB subnet group are listed.

ⓘ After a database is created, you can't change its VPC.

**DB Subnet group** [Info](#)  
Choose the DB subnet group. The DB subnet group defines which subnets and IP ranges the DB cluster can use in the VPC that you selected.

default ▼

**Public access** [Info](#)

☒ **Yes**  
RDS assigns a public IP address to the cluster. Amazon EC2 instances and other resources outside of the VPC can connect to your cluster. Resources inside the VPC can also connect to the cluster. Choose one or more VPC security groups that specify which resources can connect to the cluster.

☐ **No**  
RDS doesn't assign a public IP address to the cluster. Only Amazon EC2 instances and other resources inside the VPC can connect to your cluster. Choose one or more VPC security groups that specify which resources can connect to the cluster.

**VPC security group (firewall)** [Info](#)  
Choose one or more VPC security groups to allow access to your database. Make sure that the security group rules allow the appropriate incoming traffic.

☒ **Choose existing**  
Choose existing VPC security groups

☐ **Create new**  
Create new VPC security group

**Existing VPC security groups**

Choose one or more options ▼

default ✕

**RDS Proxy**  
RDS Proxy is a fully managed, highly available database proxy that improves application scalability, resiliency, and security.

☐ **Create an RDS Proxy** [Info](#)  
RDS automatically creates an IAM role and a Secrets Manager secret for the proxy. RDS Proxy has additional costs. For more information, see [Amazon RDS Proxy pricing](#).



► **Additional configuration**



- Leave the rest of the configuration as default and click on the create Database.

### Monitoring

#### Performance Insights [Info](#)

 Enabling Performance Insights will automatically enable the Aurora MySQL performance schema.  
[Learn more](#) 

☒ Turn on Performance Insights [Info](#)

Retention period [Info](#)


7 days (free tier) ▼

AWS KMS key [Info](#)

(default) aws/rds ▼

Account  
228078310322

KMS key ID  
d17d1fd0-5da3-463d-b05e-5304bd9afd00


 You can't change the KMS key after enabling Performance Insights.

► **Additional configuration**

Enhanced Monitoring

► **Additional configuration**


Database options, encryption turned on, failover, backup turned on, backtrack turned off, maintenance, CloudWatch Logs, delete protection turned on.

 You are responsible for ensuring that you have all of the necessary rights for any third-party products or services that you use with AWS services.

Cancel

Create database

- Amazon Elastic Container Registry
  - Navigate to Amazon Elastic Container Registry and click on Create repository button.




View push commands

Delete

Actions ▾

Create repository

< 1 > 

- Enter the name of the repository and click on Create repository.

### General settings

Visibility settings | Info


Choose the visibility setting for the repository.

☒ Private  
Access is managed by IAM and repository policy permissions.

☐ Public  
Publicly visible and accessible for image pulls.

Repository name  
Provide a concise name. A developer should be able to identify the repository contents by the name.


228078310322.dkr.ecr.us-west-2.amazonaws.com/

 Repository name is required  
0 out of 256 characters maximum (2 minimum). The name must start with a letter and can only contain lowercase letters, numbers, hyphens, underscores, periods and forward slashes.


Tag immutability | Info

Enable tag immutability to prevent image tags from being overwritten by subsequent image pushes using the same tag. Disable tag immutability to allow image tags to be overwritten.

☐ Disabled

 Once a repository is created, the visibility setting of the repository can't be changed.

### Image scan settings

 Deprecation warning  
ScanOnPush configuration at the repository level is deprecated in favor of registry level scan filters.


Scan on push  
Enable scan on push to have each image automatically scanned after being pushed to a repository. If disabled, each image scan must be manually started to get scan results.

☐ Disabled

### Encryption settings

KMS encryption  
You can use AWS Key Management Service (KMS) to encrypt images stored in this repository, instead of using the default encryption settings.

☐ Disabled

 The KMS encryption settings cannot be changed or disabled after the repository is created.

Cancel

Create repository

**Note:** Copy the URI of the repository, this will be used in the CloudIO Setup.

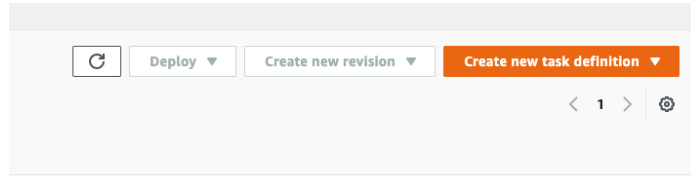
- Amazon Elastic Container Service

**Note: Please complete the CloudIO Setup before this step.**

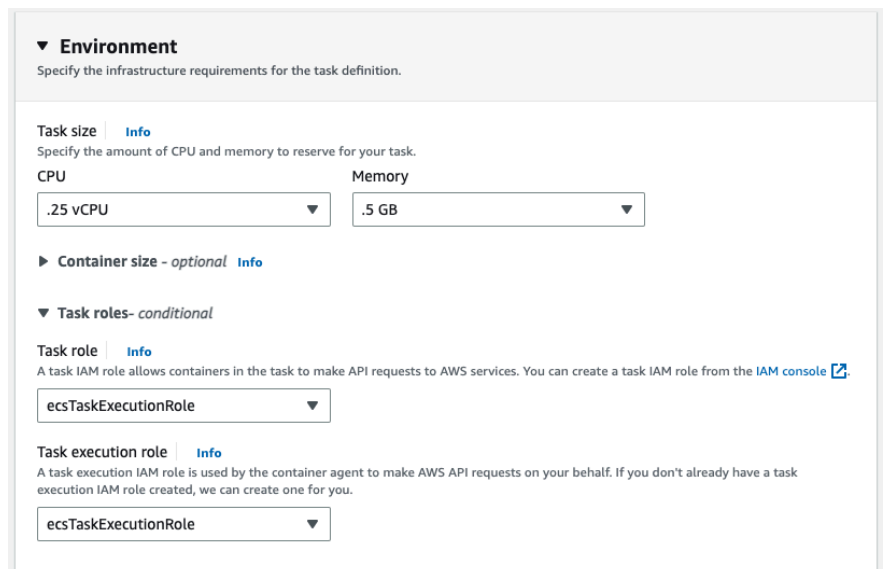
- Task Definitions

- UI

- Navigate to the Amazon Elastic Container Service -> Task Definitions. Click on Create new Task Definition.

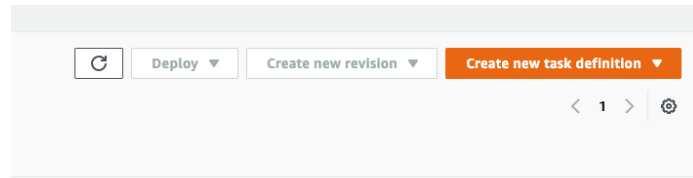


- Enter the task definition name as **ui**. Under container 1 section, enter the name of the container of your choice, Image URI as the Repository image URI in the Amazon ECR Repository, Container port as 3090, Protocol as TCP and Environment Variables as below.
  - SCHEDULER : false
  - API : true
  - STORAGE\_REGION : region where cloudio would be deployed.
  - WORKFLOW : false
  - INSTANCE\_ID : unique identifier of your choice.
  - MT\_DATABASE\_HOST\_PORT : DB URL of the Aurora MySQL which was created for CloudIO.
- Choose 0.25 CPU and 0.5 GB Memory in the environment section and click on Create.

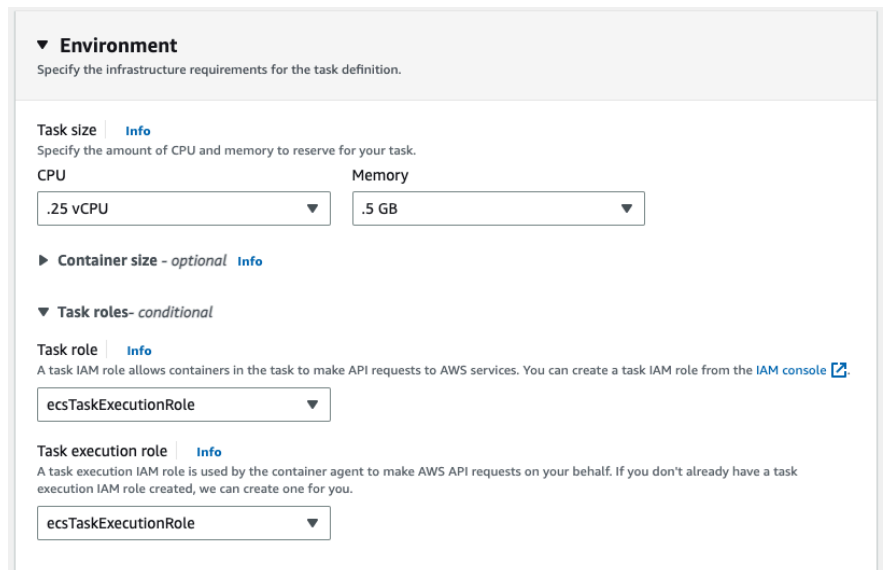
A screenshot of the 'Environment' section in the Amazon ECS console. It shows options for 'Task size' (CPU and Memory) and 'Task roles'. The 'Task size' section has dropdowns for 'CPU' (set to '.25 vCPU') and 'Memory' (set to '.5 GB'). Below this is a section for 'Task roles' with a dropdown set to 'ecsTaskExecutionRole'. There are also links for 'Info' and 'IAM console'.

- Scheduler

- Navigate to the Amazon Elastic Container Service -> Task Definitions. Click on Create new Task Definition.

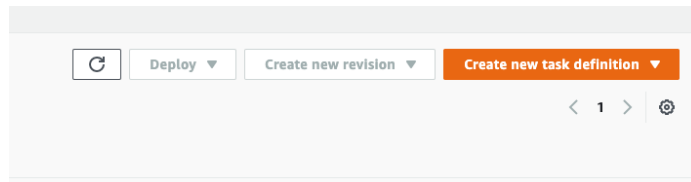


- Enter the task definition name as **scheduler**. Under container 1 section, enter the name of the container of your choice, Image URI as the Repository image URI in the Amazon ECR Repository, Container port as 3090, Protocol as TCP and Environment Variables as below.
  - SCHEDULER : true
  - API : false
  - WORKFLOW : false
  - INSTANCE\_ID : unique identifier of your choice.
- Choose 0.25 CPU and 0.5 GB Memory in the environment section and click on Create.

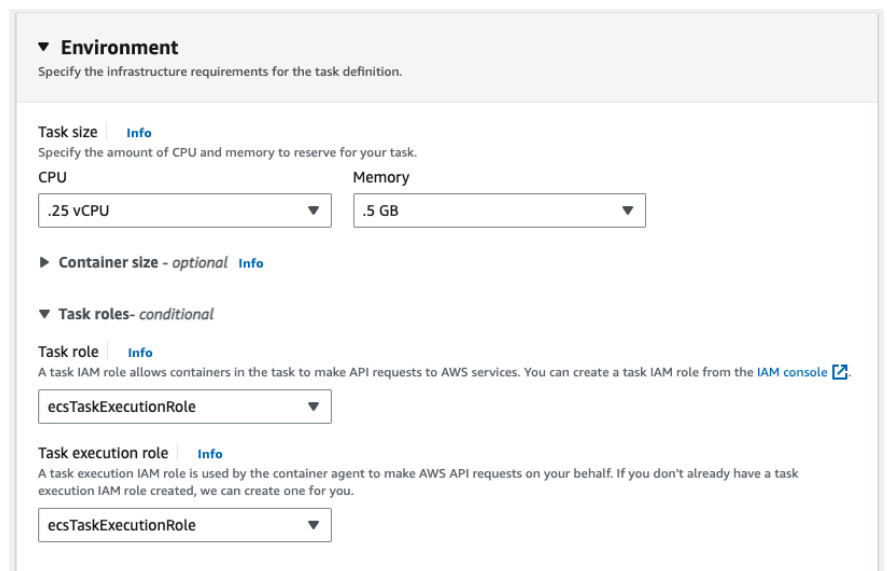
A screenshot of the 'Environment' section in the Amazon ECS console. The section is titled '▼ Environment' and includes a subtitle 'Specify the infrastructure requirements for the task definition.' It contains several configuration options: 'Task size' with a link to 'Info' and a subtitle 'Specify the amount of CPU and memory to reserve for your task.'; 'CPU' set to '.25 vCPU' and 'Memory' set to '.5 GB'; 'Container size - optional' with a link to 'Info'; 'Task roles- conditional' with a link to 'Info'; 'Task role' set to 'ecsTaskExecutionRole' with a link to 'Info' and a subtitle 'A task IAM role allows containers in the task to make API requests to AWS services. You can create a task IAM role from the IAM console'; and 'Task execution role' set to 'ecsTaskExecutionRole' with a link to 'Info' and a subtitle 'A task execution IAM role is used by the container agent to make AWS API requests on your behalf. If you don't already have a task execution IAM role created, we can create one for you.'

- Workflow

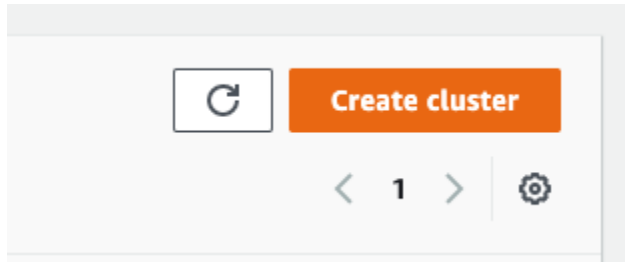
- Navigate to the Amazon Elastic Container Service -> Task Definitions. Click on Create new Task Definition.



- Enter the task definition name as **workflow**. Under container 1 section, enter the name of the container of your choice, Image URI as the Repository image URI in the Amazon ECR Repository, Container port as 3090, Protocol as TCP and Environment Variables as below.
  - SCHEDULER : false
  - API : false
  - WORKFLOW : true
  - STORAGE\_REGION: Region where CloudIO would be deployed.
  - INSTANCE\_ID : unique identifier of your choice.
- Choose 0.25 CPU and 0.5 GB Memory in the environment section and click on Create.



- - Cluster
    - Navigate to the Amazon Elastic Container Service ->Cluster and click on Create Cluster button.



- Choose cluster name, VPC and subnets to which CloudIO would have access to, AWS Fargate Infrastructure and click on Create button.

Cluster name

Cluster name here. For example, DevCluster.

There can be a maximum of 255 characters. The valid characters are letters (uppercase and lowercase), numbers, hyphens, and underscores.

▼ Networking [Info](#)

By default tasks and services run in the default subnets for your default VPC. To use the non-default VPC, specify the VPC and subnets.

VPC

Use a VPC with public and private subnets. By default, VPCs are created for your AWS account. To create a new VPC, go to the [VPC Console](#).

vpc-72478716

CloudIO | default

Subnets

Select the subnets where your tasks run. We recommend that you use three subnets for production.

Choose subnets

subnet-79c60d1d

us-west-2b

subnet-d91f2780

us-west-2c

subnet-162ec260

us-west-2a

Default namespace - optional

Select the namespace to specify a group of services that make up your application. You can overwrite this value at the service level.

Specify a namespace

▼ Infrastructure [Info](#)

Serverless

Your cluster is automatically configured for AWS Fargate (serverless) with two capacity providers. Add Amazon EC2 instances, or external instances using ECS Anywhere.

☒ AWS Fargate (serverless)

Pay as you go. Use if you have tiny, batch, or burst workloads or for zero maintenance overhead. The cluster has Fargate and Fargate Spot capacity providers by default.

☐ Amazon EC2 instances

Manual configurations. Use for large workloads with consistent resource demands.

☐ External instances using ECS Anywhere

Manual configurations. Use to add data center compute.

► Monitoring - optional [Info](#)

Container Insights is off by default. When you use Container Insights, there is a cost associated with it.

► Tags - optional [Info](#)

Tags help you to identify and organize your clusters.

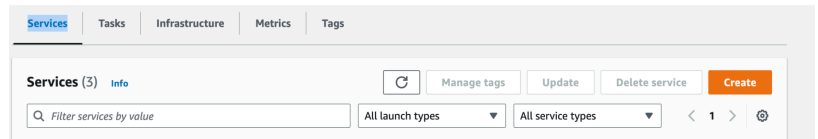
Cancel

Create



- Services
  - *UI*

- Navigate to cluster created above and click on Create button at the Services tab



- Choose *ui* as family and revision as latest and service name as *ui* in the deployment configuration section.

### Deployment configuration

Application type [Info](#)  
Specify what type of application you want to run.

☒ Service  
Launch a group of tasks handling a long-running computing work that can be stopped and restarted. For example, a web application.

☐ Task  
Launch a standalone task that runs and terminates. For example, a batch job.

Task definition  
Select an existing task definition. To create a new task definition, go to [Task definitions](#).

☐ Specify the revision manually  
Manually input the revision instead of choosing from the 100 most recent revisions for the selected task definition family.

Family  
ui

Revision  
2 (LATEST)

Service name  
Assign a unique name for this service.

ui

Service type [Info](#)  
Specify the service type that the service scheduler will follow.

☒ Replica  
Place and maintain a desired number of tasks across your cluster.

☐ Daemon  
Place and maintain one copy of your task on each container instance.

Desired tasks  
Specify the number of tasks to launch.

1

► Deployment options

► Deployment failure detection [Info](#)

- Choose the VPC, Subnets and Security group to which CloudIO will have access to in the networking section.

▼ Networking

VPC [Info](#)  
Choose the Virtual Private Cloud to use.  
vpc-72478716  
CloudIO | default

Subnets  
Choose the subnets within the VPC that the task scheduler should consider for placement.  
Choose subnets

subnet-79c60d1d × subnet-d91f2780 × subnet-162ec260 ×  
us-west-2b us-west-2c us-west-2a

Security group [Info](#)  
Choose an existing security group or create a new security group.  
☒ Use an existing security group  
☐ Create a new security group  
Security group name  
Choose an existing security group.

sg-09a8c06e ×  
default | default

Public IP [Info](#)  
Choose whether to auto-assign a public IP to the task's elastic network interface (ENI).

☒ Turned on

- Use the load balancer, target group created in the Load balancer section and click on create button.

▼ Load balancing - optional

Load balancer type [Info](#)

Configure a load balancer to distribute incoming traffic across the tasks running in your service.

Application Load Balancer

Application Load Balancer

Specify whether to create a new load balancer or choose an existing one.

☐ Create a new load balancer

☒ Use an existing load balancer

Load balancer

Select the load balancer you wish to use to distribute incoming traffic across the tasks running in your service.

xorapps-lb-v2

Choose container to load balance

xorapps 3090:3090

Listener [Info](#)

Specify the port and protocol that the load balancer will listen for connection requests on.

Port

80

Protocol

HTTP

Target group [Info](#)

Specify whether to create a new target group or choose an existing one that the load balancer will use to route requests to the tasks in your service.

☐ Create new target group

☒ Use an existing target group

Target group name

xorapps-tg-v2

Health check path

/ping

Health check protocol

HTTP

Health check grace period [Info](#)

seconds

## ○ Scheduler

- Navigate to cluster created above and click on Create button at the Services tab

Services | Tasks | Infrastructure | Metrics | Tags

Services (3) [Info](#)

Refresh

Manage tags

Update

Delete service

Create

Filter services by value

All launch types

All service types

< 1 > ⚙

©2023 CloudIO-ALL RIGHTS RESERVED

- Choose *scheduler* as family and revision as latest and service name as *scheduler* in the deployment configuration section.

### Deployment configuration

Application type [Info](#)  
Specify what type of application you want to run.

☒ **Service**  
Launch a group of tasks handling a long-running computing work that can be stopped and restarted. For example, a web application.

☐ **Task**  
Launch a standalone task that runs and terminates. For example, a batch job.

Task definition  
Select an existing task definition. To create a new task definition, go to [Task definitions](#).

☐ **Specify the revision manually**  
Manually input the revision instead of choosing from the 100 most recent revisions for the selected task definition family.

Family  
scheduler

Revision  
1 (LATEST)

Service name  
Assign a unique name for this service.

scheduler

Service type [Info](#)  
Specify the service type that the service scheduler will follow.

☒ **Replica**  
Place and maintain a desired number of tasks across your cluster.

☐ **Daemon**  
Place and maintain one copy of your task on each container instance.

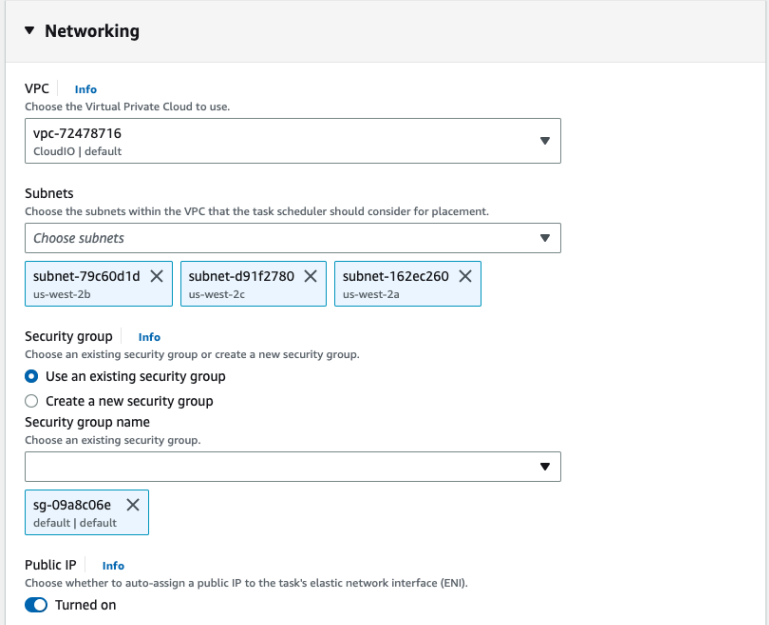
Desired tasks  
Specify the number of tasks to launch.

1

► Deployment options

► Deployment failure detection [Info](#)

- Choose the VPC, Subnets and Security group to which CloudIO will have access to in the networking section.



**Networking**

**VPC** [Info](#)  
Choose the Virtual Private Cloud to use.  
vpc-72478716  
CloudIO | default

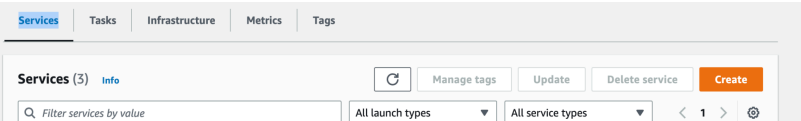
**Subnets**  
Choose the subnets within the VPC that the task scheduler should consider for placement.  
Choose subnets  
subnet-79c60d1d us-west-2b subnet-d91f2780 us-west-2c subnet-162ec260 us-west-2a

**Security group** [Info](#)  
Choose an existing security group or create a new security group.  
☒ Use an existing security group  
☐ Create a new security group  
Security group name  
Choose an existing security group.  
sg-09a8c06e default | default

**Public IP** [Info](#)  
Choose whether to auto-assign a public IP to the task's elastic network interface (ENI).  
☒ Turned on

○ Workflow

- Navigate to cluster created above and click on Create button at the Services tab



**Services** [Info](#)

Services (3) [Manage tags](#) [Update](#) [Delete service](#) [Create](#)

Filter services by value All launch types All service types

- Choose *workflow* as family and revision as latest and service name as *workflow* in the deployment configuration section.

### Deployment configuration

Application type [Info](#)  
Specify what type of application you want to run.

☒ **Service**  
Launch a group of tasks handling a long-running computing work that can be stopped and restarted. For example, a web application.

☐ **Task**  
Launch a standalone task that runs and terminates. For example, a batch job.

Task definition  
Select an existing task definition. To create a new task definition, go to [Task definitions](#).

☐ **Specify the revision manually**  
Manually input the revision instead of choosing from the 100 most recent revisions for the selected task definition family.

Family  
workflow

Revision  
3 (LATEST)

Service name  
Assign a unique name for this service.

workflow

Service type [Info](#)  
Specify the service type that the service scheduler will follow.

☒ **Replica**  
Place and maintain a desired number of tasks across your cluster.

☐ **Daemon**  
Place and maintain one copy of your task on each container instance.

Desired tasks  
Specify the number of tasks to launch.

1

► Deployment options

► Deployment failure detection [Info](#)

- Choose the VPC, Subnets and Security group to which CloudIO will have access to in the networking section.

▼ Networking

VPC

Info

Choose the Virtual Private Cloud to use.

vpc-72478716

CloudIO | default

Subnets

Choose the subnets within the VPC that the task scheduler should consider for placement.

Choose subnets

subnet-79c60d1d

×

us-west-2b

subnet-d91f2780

×

us-west-2c

subnet-162ec260

×

us-west-2a

Security group

Info

Choose an existing security group or create a new security group.

☒ Use an existing security group

☐ Create a new security group

Security group name

Choose an existing security group.

sg-09a8c06e

×

default | default

Public IP

Info

Choose whether to auto-assign a public IP to the task's elastic network interface (ENI).

☒ Turned on

## ● CloudIO Setup

- Once you obtain a license from CloudIO, follow the instructions to download cloudio-platform.zip and unzip to a directory and update the .env file with appropriate values for the following environment variables

Environment Variable	Description
API	Set it to true to enable the API Service (UI Backend)
SCHEDULER	Set it to true to enable the scheduler service
WORKFLOW	Set it to true to enable the workflow service
IO_ENV	development/test/production
LOG_OUTPUT	file or console

REDIS_URL	URL of the redis instance installed above
JWT_SECRET	Used to encode/decode JWT tokens
ARGON_SECRET	Used for password hashing
DATABASE_URL	MySQL database URL. Make sure to include the schema name cloudio as well as part of the URL. Refer to the sample below.
READONLY_DATABASE_URL	Used for running ad hoc queries from SQL Worksheet
DB_ROOT_CERT_PATH	CA cert path
DB_PKCS12_PATH	Private key in PKCS12 format
DB_PKCS12_PASSWORD	Private key password if any
DB_ACCEPT_INVALID_CERTS	To accept invalid certs (self signed certs)
DB_SKIP_DOMAIN_VALIDATION	To skip domain validation
BOOTSTRAP_SERVERS	Kafka bootstrap server URL. If using a cloud instance from confluent then provide appropriate values for the additional variables SECURITY_PROTOCOL, SASL_MECHANISMS, SASL_USERNAME & SASL_PASSWORD provided



	by confluent cloud when creating a new kafka cluster
INSTANCE_ID	A unique name for this instance
HOST	An IP address and port combination on which the web server listens for incoming connections. You can run multiple instances on the same host with different ports and/or on multiple hosts depending on the load. A single instance can scale upto a million requests per 20 minutes.
API_RATELIMIT	Number of API calls allowed per IP address per hour
SMTP_HOST	SMTP Host Name to be used for sending email alerts
SMTP_USERNAME	SMTP Username
SMTP_PASSWORD	SMTP Password
SMTP_FROM	From email address to be used for the outbound emails
STORAGE_ACCESS_KEY	Minio/S3 Access Key
STORAGE_SECRET_KEY	Minio/S3 Secret Key
TMP_DIR	Temp directory path
ALLOW_SQL_WORKSHEET_UPDATES	

- Once extracted, navigate to the cloudio folder from terminal and run the pull.sh file to get the latest CloudIO build.

```
cd cloudio
```

```
./pull.sh
```

- After pulling the latest build, push the image to the amazon container registry using the push commands.

- **Retrieve an authentication token and authenticate your Docker client to your registry.**

Use the AWS CLI:

```
aws ecr get-login-password --region us-west-2 | docker login --username AWS  
--password-stdin {container_registry_url}
```

Note: If you receive an error using the AWS CLI, make sure that you have the latest version of the AWS CLI and Docker installed.

- **Build your Docker image using the following command. For information on building a Docker file from scratch see the instructions [here](#) . You can skip this step if your image is already built:**

```
docker build -t {app_name} .
```

- **After the build completes, tag your image so you can push the image to this repository:**

```
docker tag {app_name}:latest {container_registry_url}/{app_name}:latest
```

- **Run the following command to push this image to your newly created AWS repository:**

```
docker push {container_registry_url}/{app_name}:latest
```