Creating an IBM Cloud Container Registry Namespace

Objectives

After completing this lab, you will be able to:

- Describe the IBM Cloud Container Registry service
- Create a Container Registry namespace

Lab Overview

In this lab you will create an IBM Cloud Container Registry namespace, which you will use in a subsequent labs.

Pre-requisites

You will need an IBM Cloud account to do this lab. If you have not created one already, click on this link and follow the instructions to create an IBM Cloud account.

About IBM Cloud

The IBM Cloud platform is deployed across data centers around the world. It combines platform as a service (PaaS) with infrastructure as a service (IaaS) to provide an integrated experience. The platform scales and supports both large enterprise businesses and small development teams and organizations.

The platform is built to support your needs, whether it's working only in the public cloud or taking advantage of a multicloud deployment model. IBM Cloud offers a variety of services, including Compute, Network, Storage, Management, Security, Databases, Analytics, AI, and Cloud Paks.

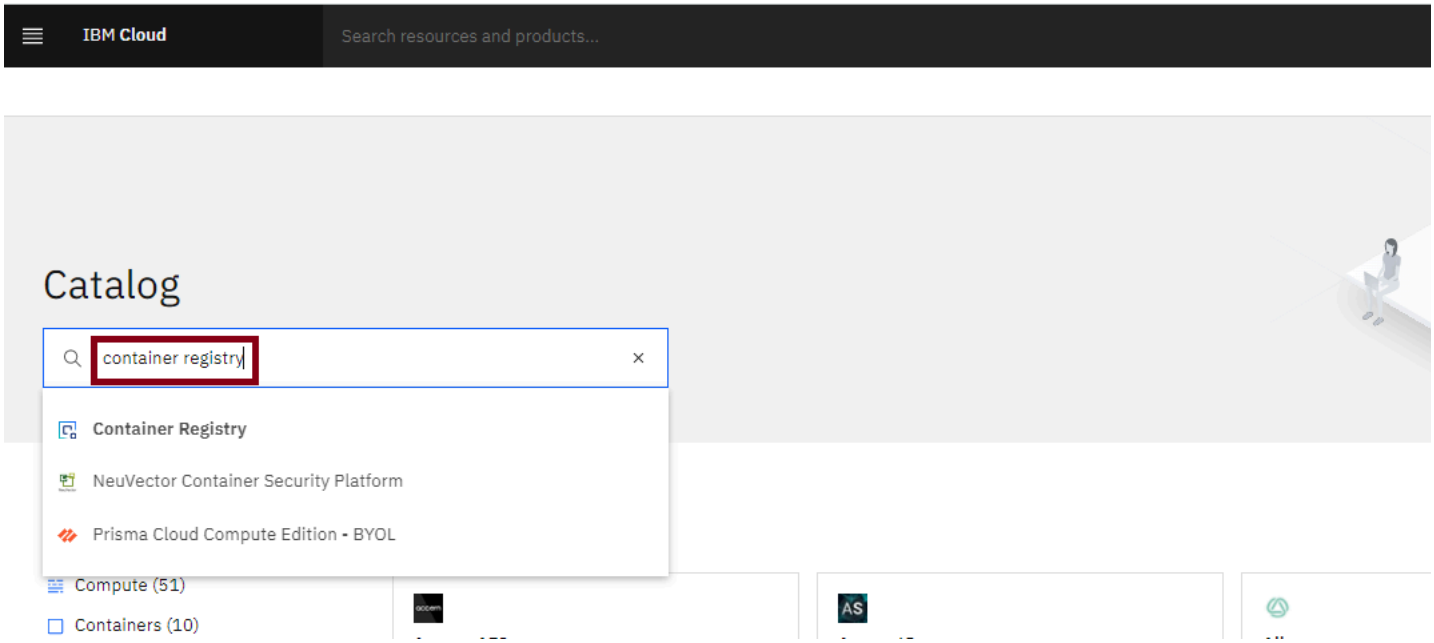
About IBM Cloud Container Registry namespaces

IBM Cloud® Container Registry provides a multi-tenant, encrypted private image registry that you can use to store and access your container images in a highly available and scalable architecture. The namespace is a slice of the registry to which you can push your images. The namespace will be a part of the image name when you tag and push an image. For example, `us.icr.io/<my_namespace>/<my_repo>:<my_tag>`.

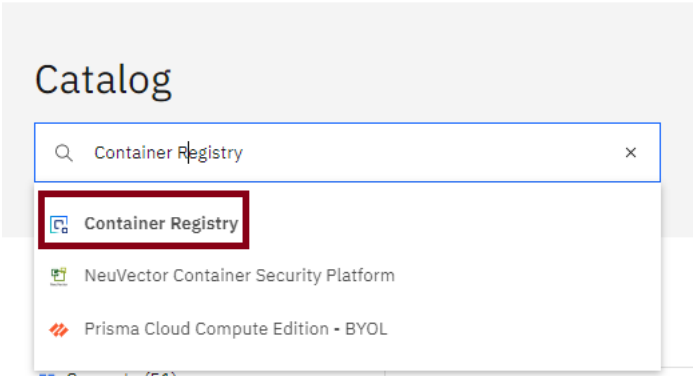
Create a Container Registry namespace

1. Go to the IBM Cloud catalog page.

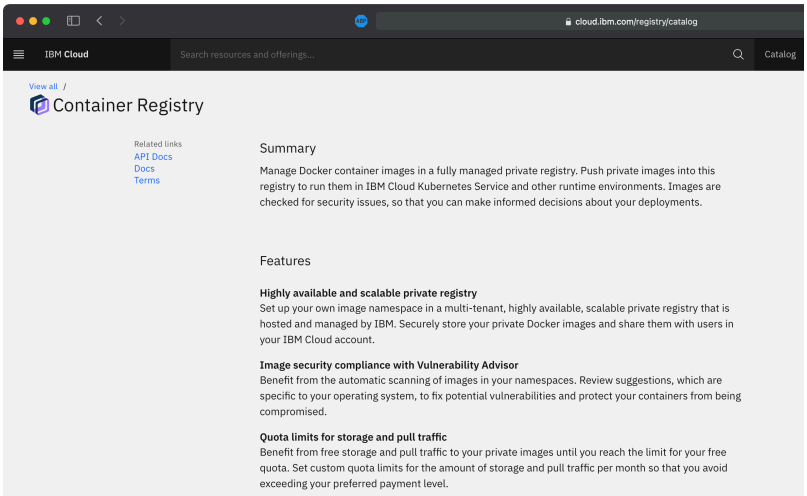
2. In the **Catalog** search box, type `Container Registry`.



3. Click on **Container Registry** in the search results.



4. You can now read about the Container Registry service and visit links for API documentation and docs about how to use the service.



5. At the top right, click **Get started**.

View all /

Container Registry

Author: IBM • Docs • API docs

Compliance
EU Supported

Related links
API Docs
Docs
Terms

Summary

Manage Docker container images in a fully managed private registry. Push private images into this registry to run them in IB
can make informed decisions about your deployments.

Features

Highly available and scalable private registry
Set up your own image namespace in a multi-tenant, highly available, scalable private registry that is hosted and managed

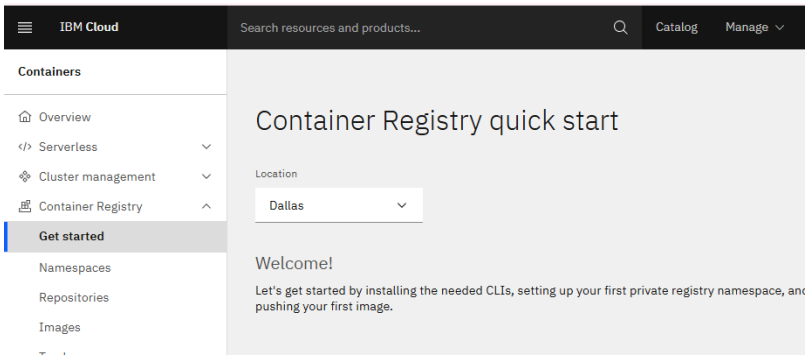
Image security compliance with Vulnerability Advisor
Benefit from the automatic scanning of images in your namespaces. Review suggestions, which are specific to your operati

Quota limits for storage and pull traffic
Benefit from free storage and pull traffic to your private images until you reach the limit for your free quota. Set custom quo
level.

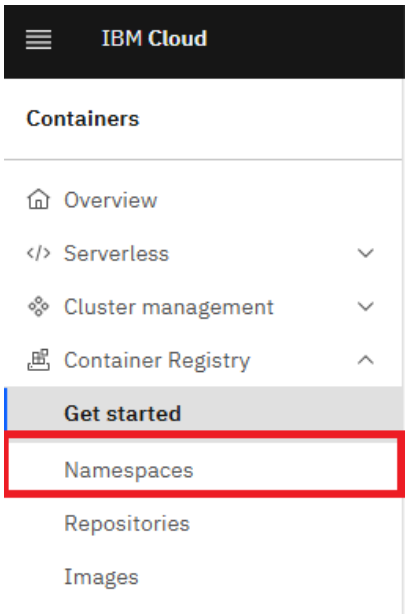
Pricing plans

Plan	Features
Lite	Free Plan with limited resources Storage (Gigabyte-Months) - 0.5 GB free per month Pull traffic (Gigabytes) - 5 GB free per month
Container Registry	Namespaces for Container Registry
Standard	Pull traffic (Gigabytes) - 5 GB free per month Storage (Gigabyte-Months) - 0.5 GB free per month The plan provides a free tier and unlimited use at a cost. You can set limits to manage your costs.

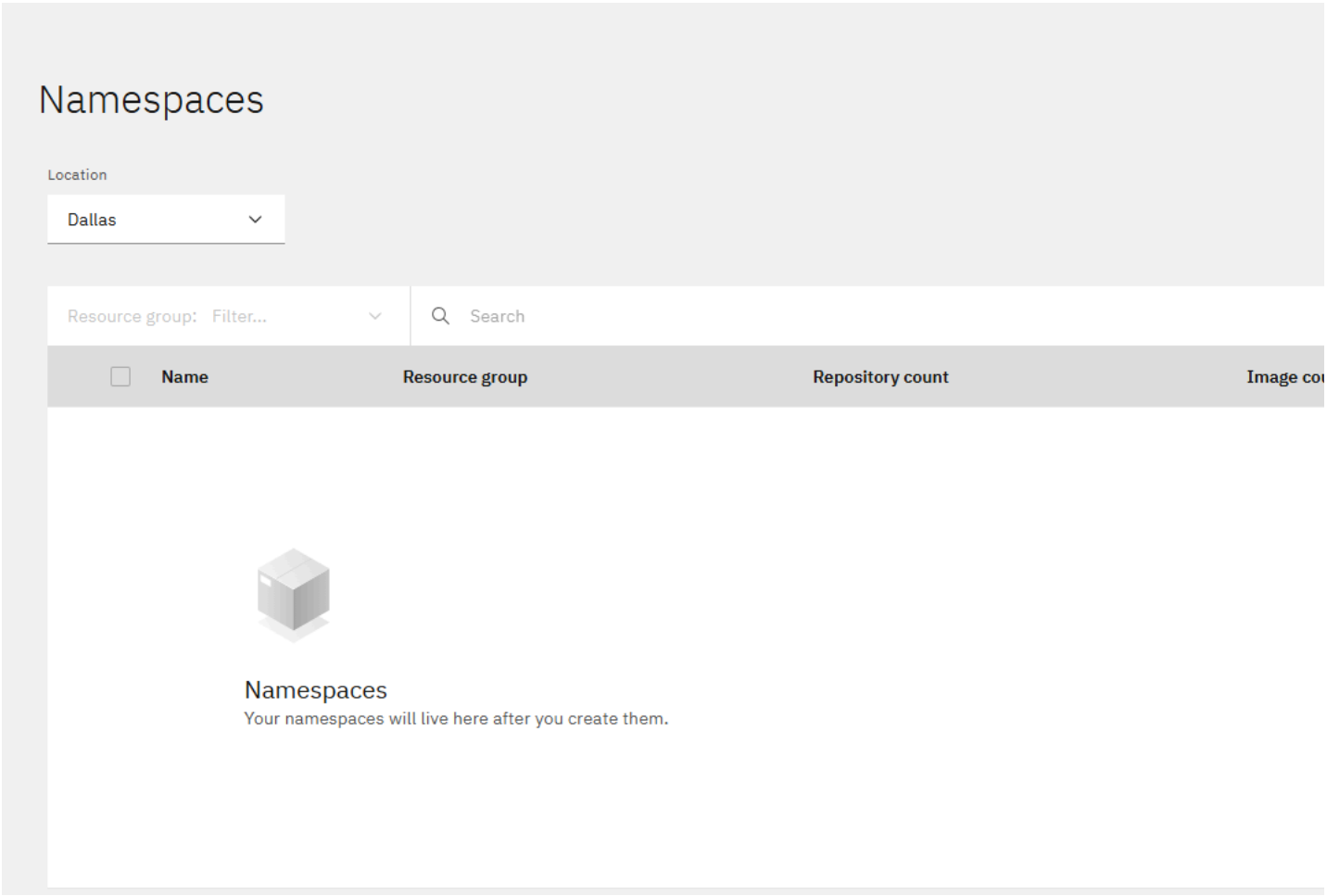
6. Ensure that the location is set to **Dallas**.



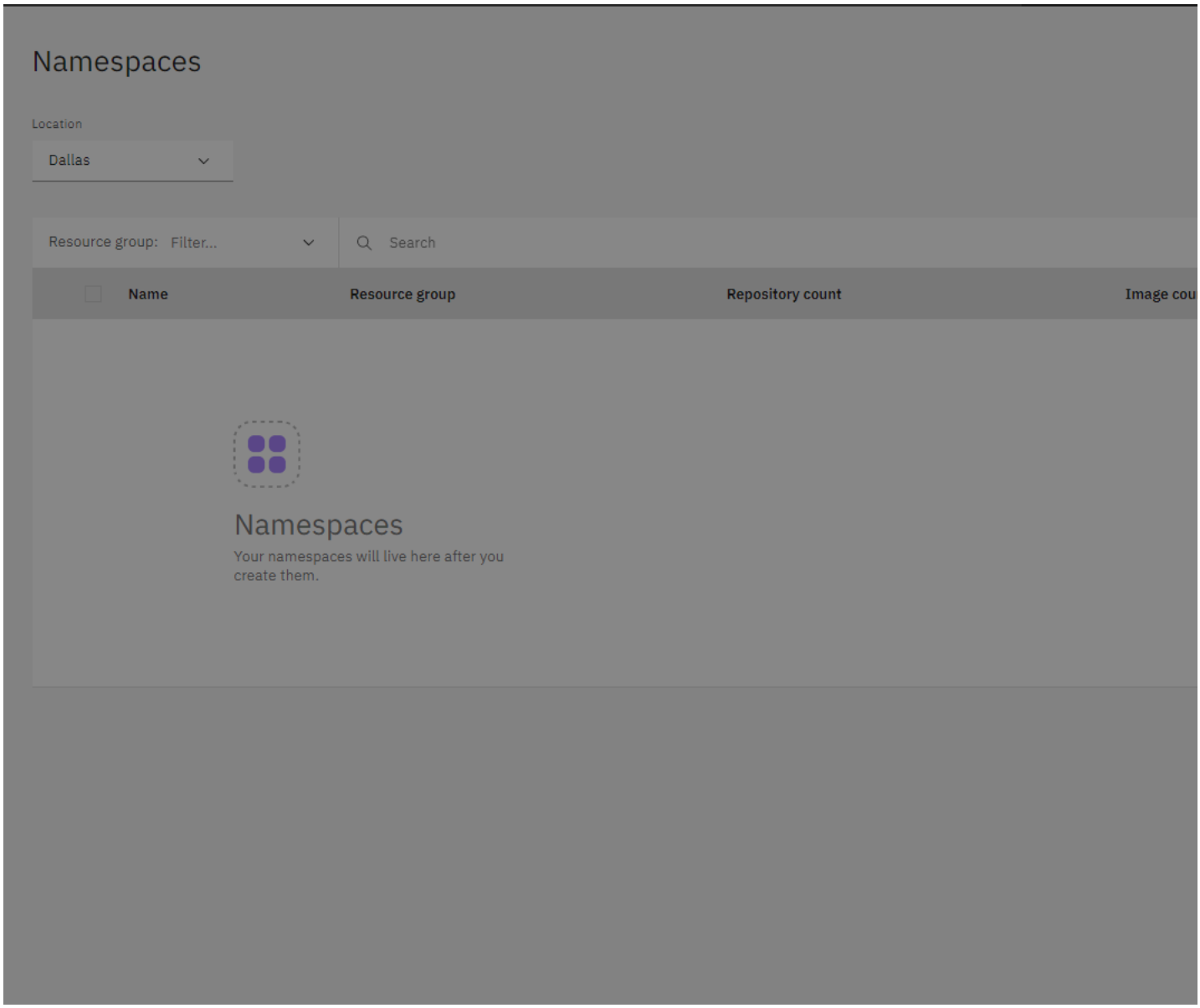
7. On the left hand side panel, click the **Namespaces** tab.



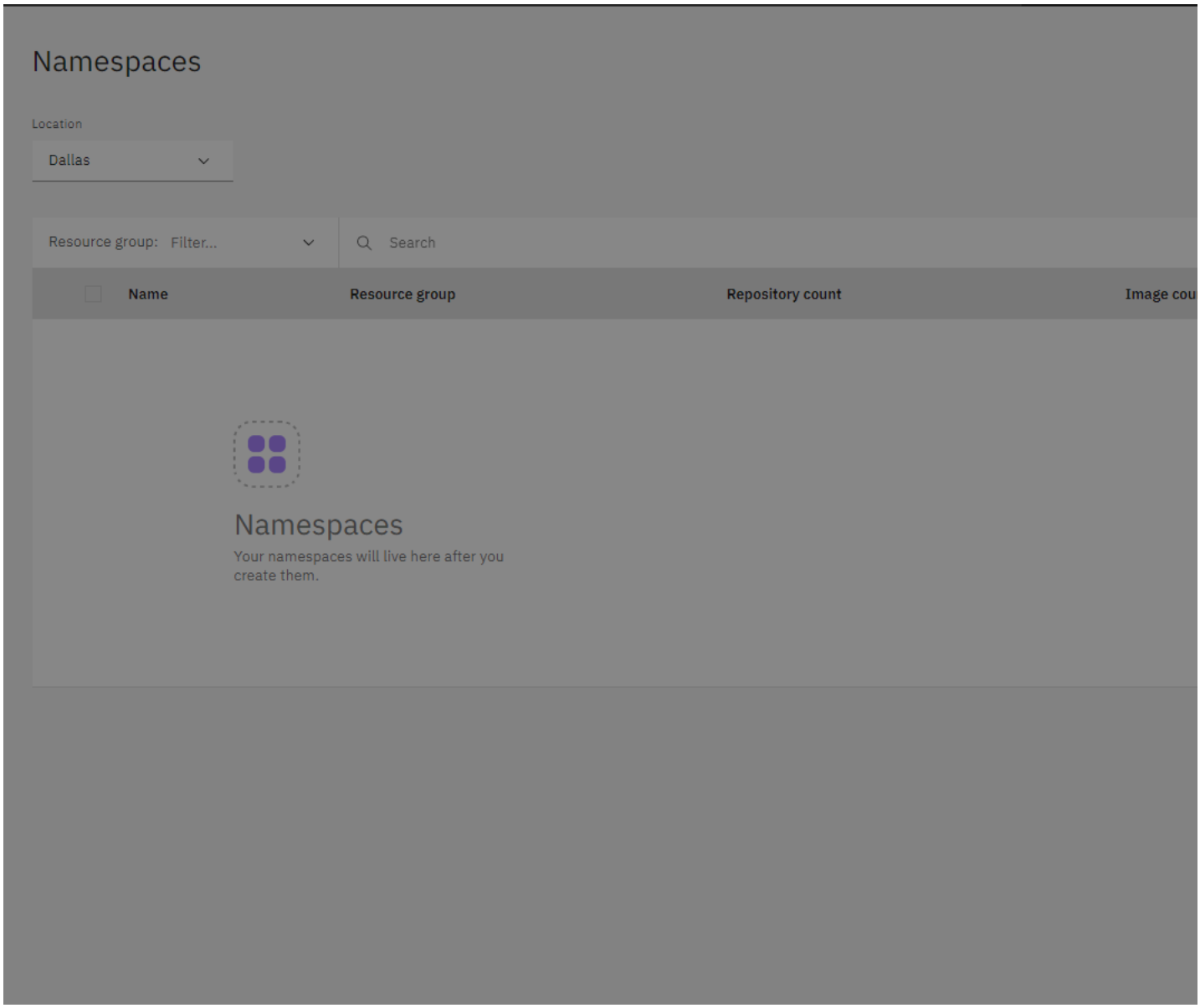
8. On the right side of the Namespaces panel, click **Create**.



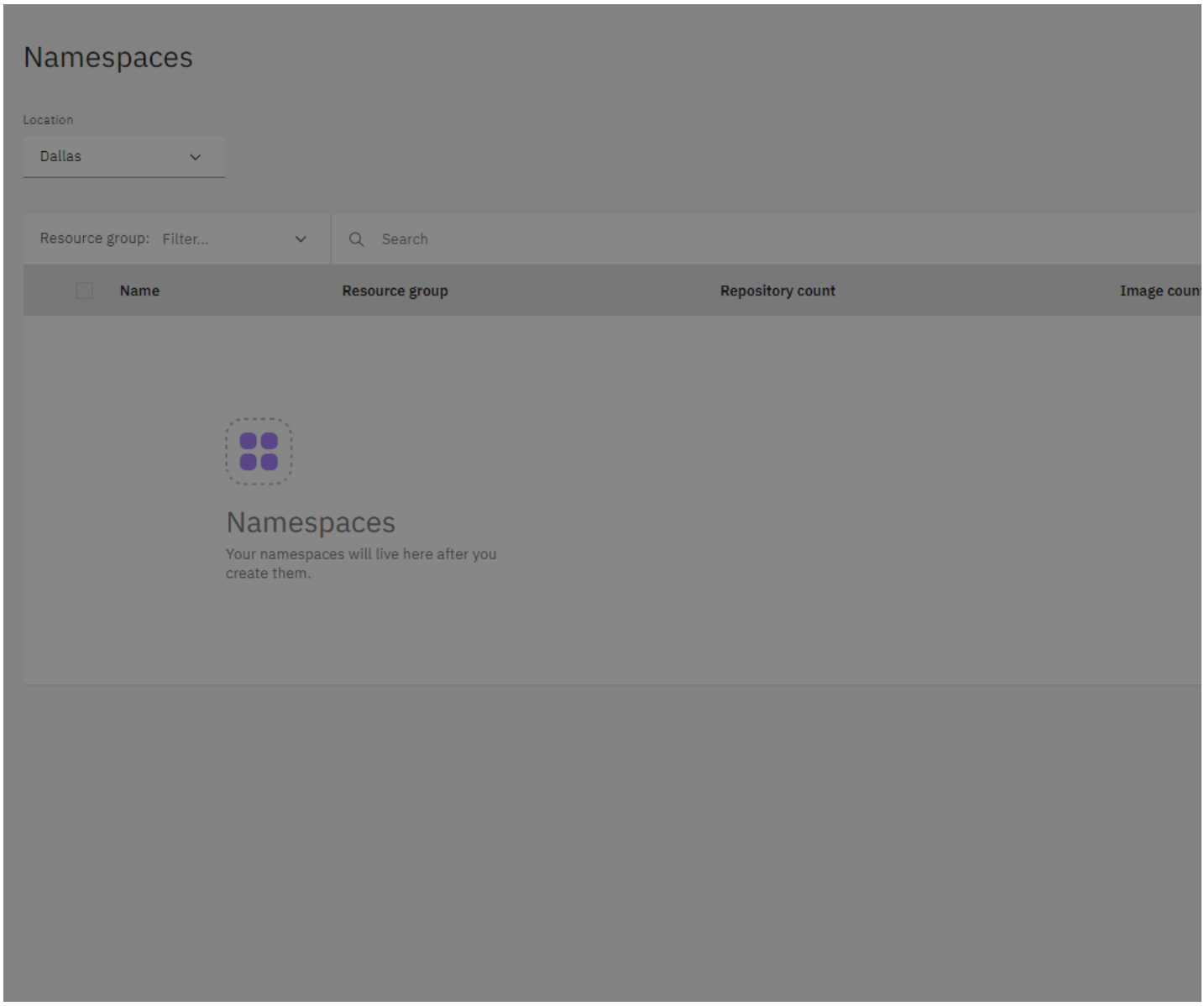
9. A **Create namespace** panel opens.



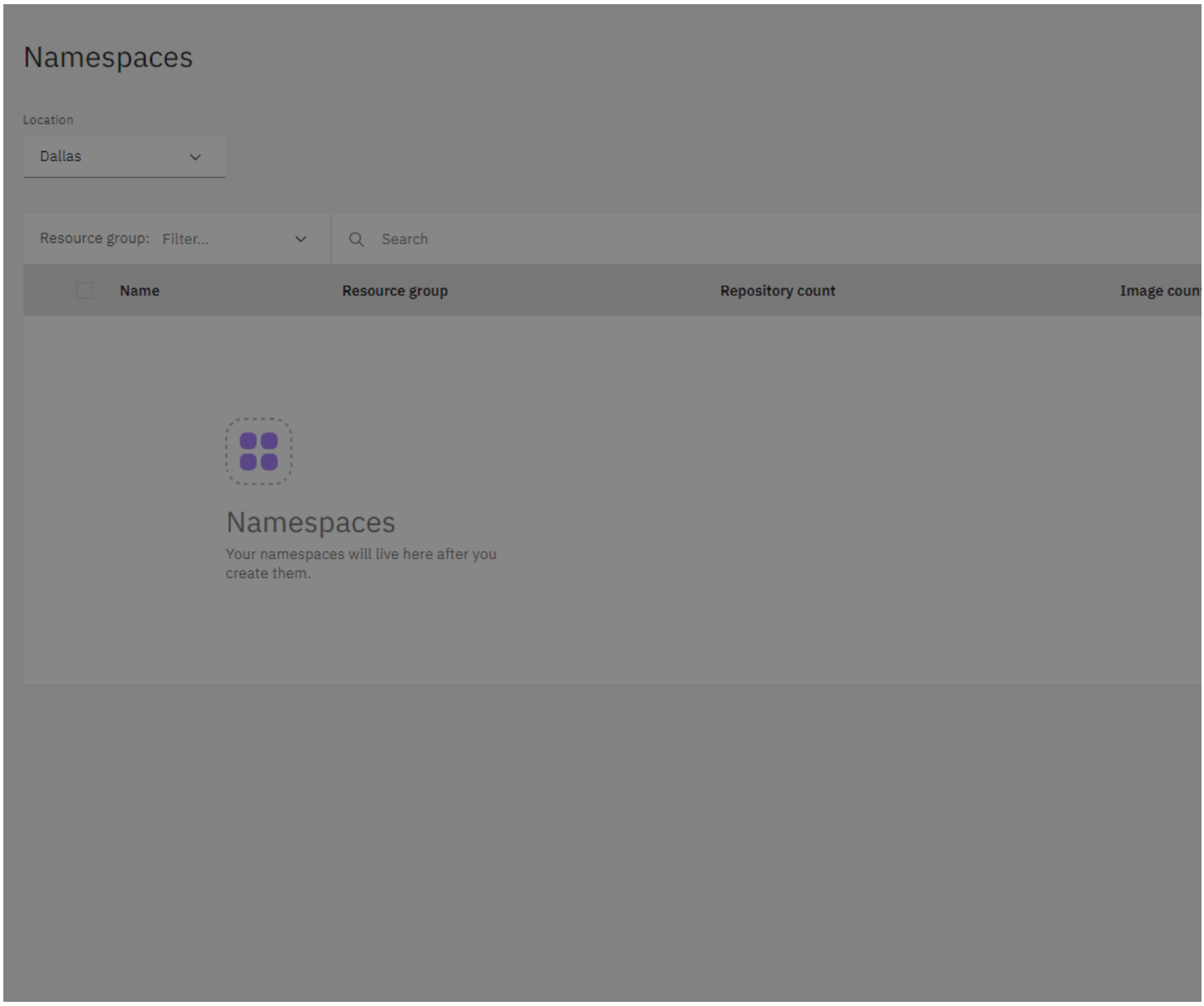
10. In the **Resource group** field, select the name of the resource group you would like this namespace to reside in. For this lab, you can simply leave the selection as **Default**.



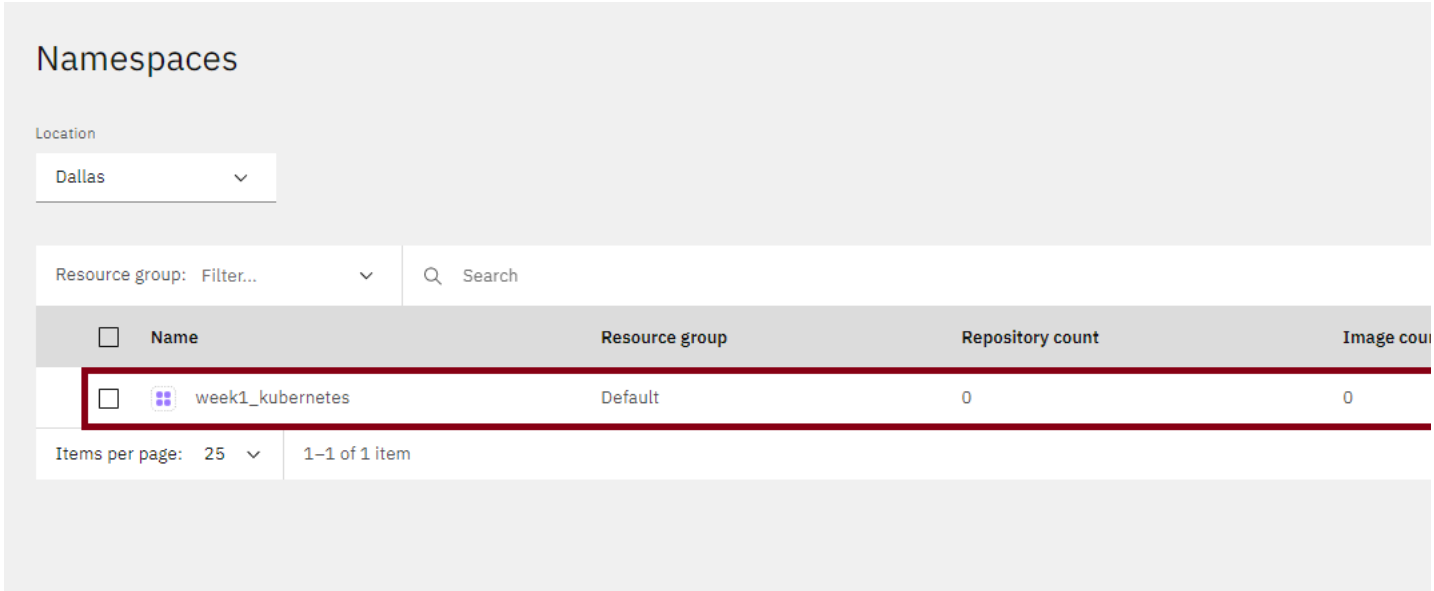
11. In the **Name** field, type a unique name for the namespace. The name must be unique across all users of the Container Registry service in this region.



12. Click **Create** at the bottom of the panel to create the namespace.



You now have a namespace (as below) to which you can push images.



Congratulations! You have completed the first lab for the first module of this course.

Skills Network