Lesson 1 : Navigate OpenShift Web Console

Navigate OpenShift Web Console

Administrative tasks

- Command Line Interface (CLI)
  - **oc** Command
  - **kubectl** Command
  - Can be scripted
- Graphical User Interface – Web Console
  - Uses Kubernetes API and OpenShift extension APIs
  - Ease of access and management

The Web Consoles

Kubernetes

- Web-based
  - Not deployed by default
  - Minimal security permissions
  - Accepts only token-based authentication
  - Require Proxy setup

OpenShift

- Web-based
  - fully deployed
  - Additional operators extend features and functionality
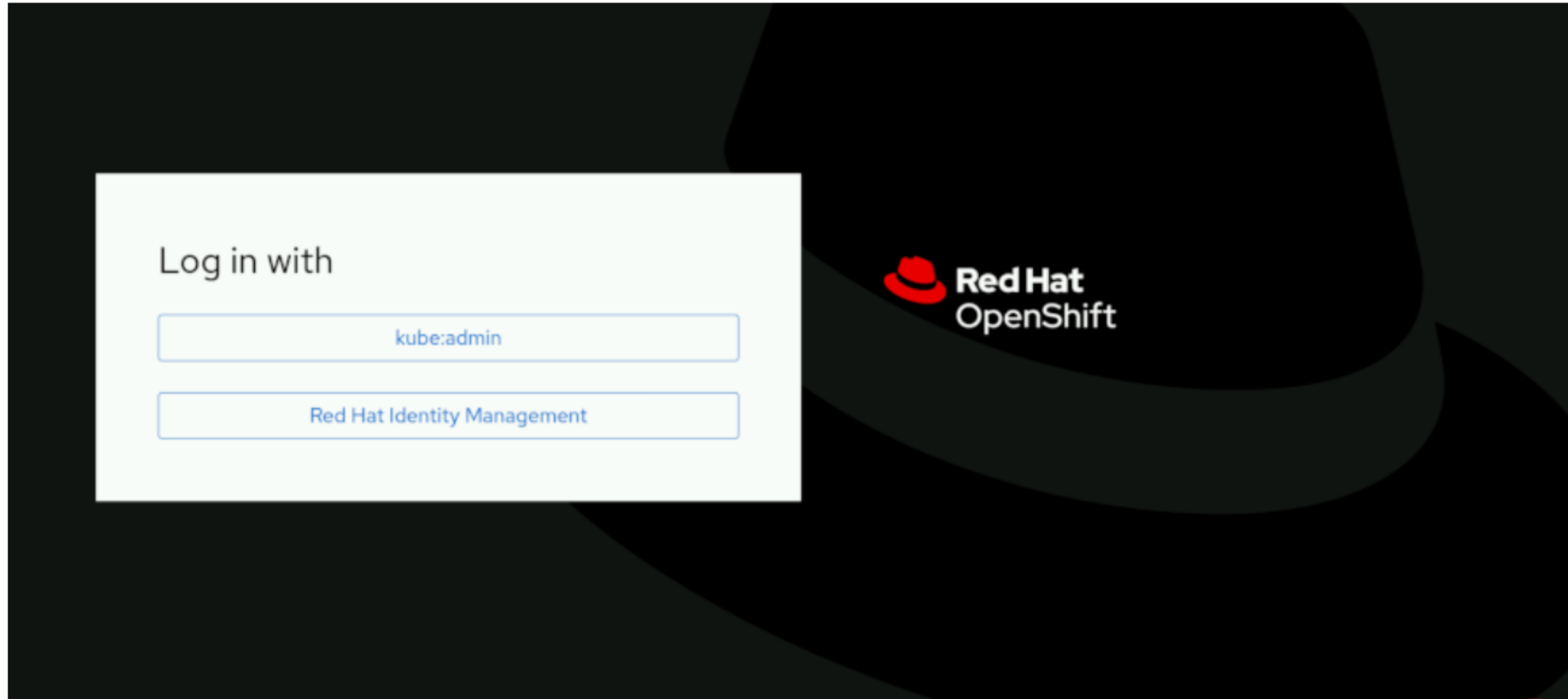
Accessing the OpenShift Web Console

```
[user@host ~]$ oc login -u user -p password https://api.ocp4.example.com:6443
Login successful.

...output omitted...
```
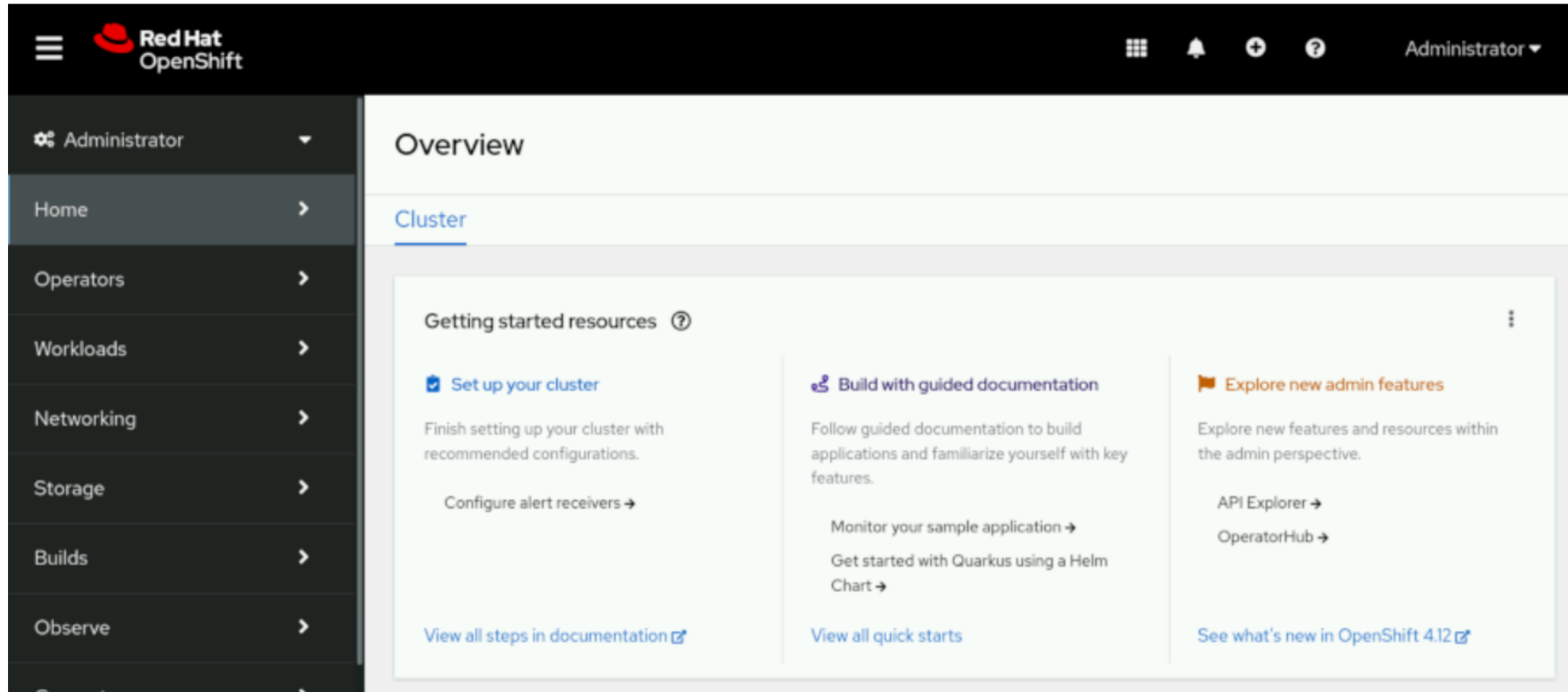
Then, you execute the `oc whoami --show-console` command to retrieve the web console URL:

```
[user@host ~]$ oc whoami --show-console
https://console-openshift-console.apps.ocp4.example.com
```
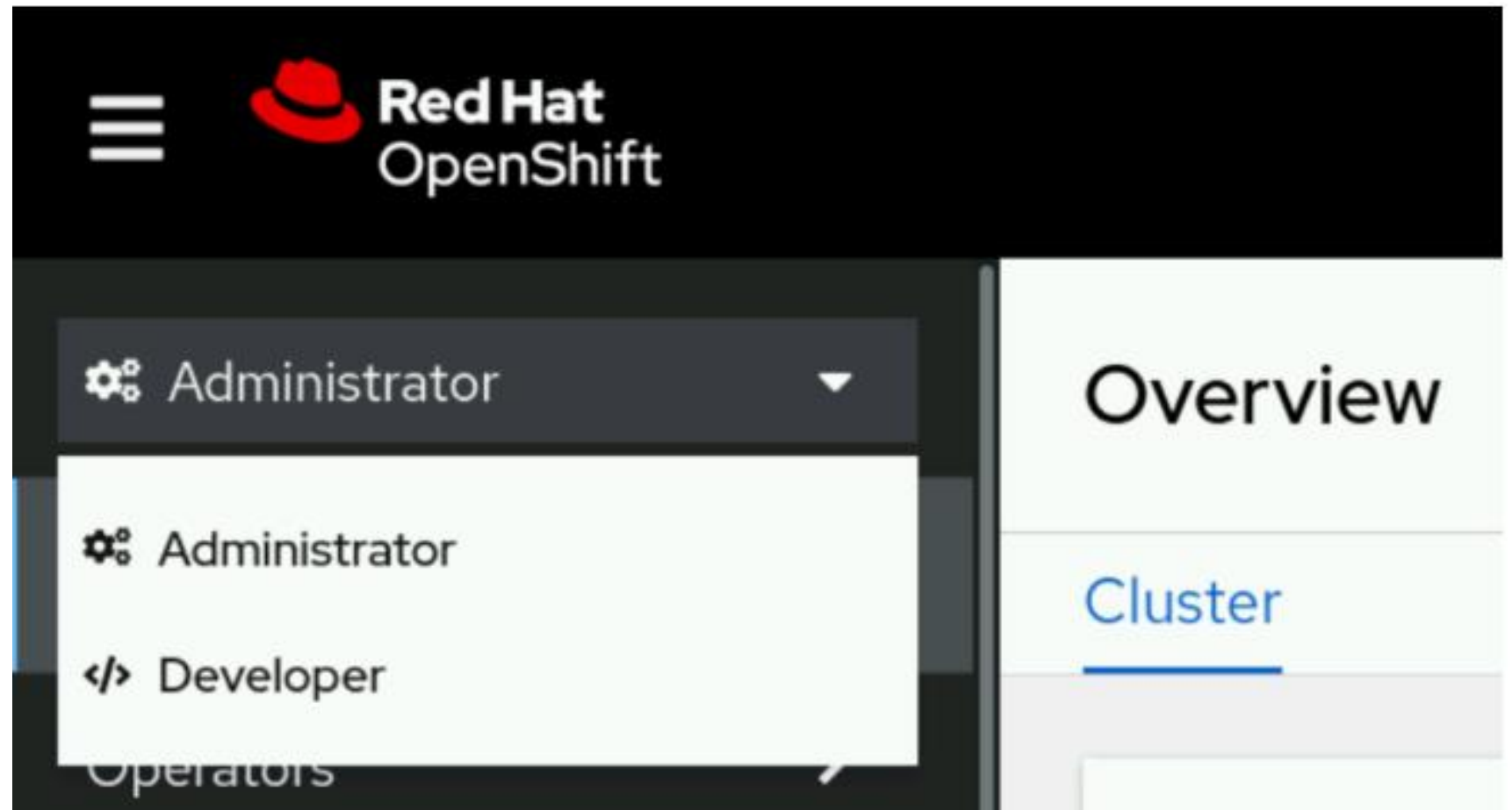
First page – The authentication

OpenShift home page

Web Console Perspective
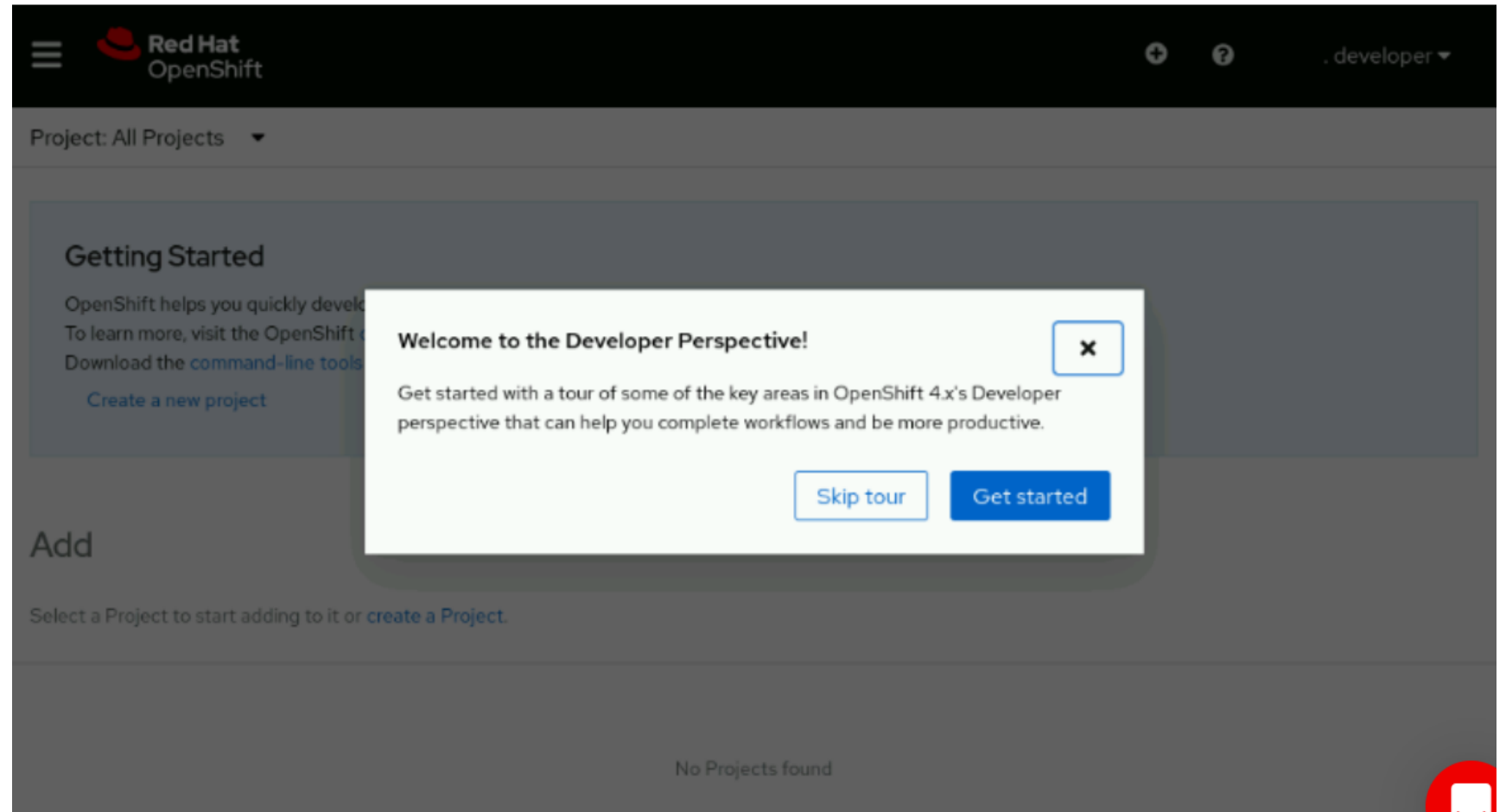

Administrator perspective
- cluster configuration
- operations of clusters
- creating deployments
- running workloads

Developer perspective
- creating deployment
- running workloads

Short information tour

Access OperatorHub web console

- Explore operators from Kubernetes community or Red Hat partners

Access operatorHub using command line

```
get packagemanifests -n openshift-marketplace
```

iple output

NAME	CATALOG
3scale-operator	Red Hat Operators
advanced-cluster-management	Red Hat Operators
amq7-cert-manager	Red Hat Operators
...	
couchbase-enterprise-certified	Certified Operators
crunchy-postgres-operator	Certified Operators
mongodb-enterprise	Certified Operators
...	
etcd	Community Operators
jaeger	Community Operators
kubefed	Community Operators

The Developer Catalog

- After installation of operators, their capabilities will appear in Developer Catalog

installation, the Operator capabilities will appear in the Developer Catalog providing a self-service experience.

All items

- CI/CD
- Databases
- Languages
- Middleware
- Other

Type ⓘ
- Builder Images (12)
- Devfiles (6)
- Helm Charts (51)
- Operator Backed (2)

All items

🔍 Filter by keyword...   A-Z ▾   116 items

Builder Images

.NET

.NET
Provided by Red Hat

Build and run .NET 7 applications on UBI 8. For more information about using this builder image,...

Helm Charts

HELM

.NET
Provided by Red Hat

A Helm chart to build and deploy .NET applications

Red Hat OpenShift Key Concepts

- Pods: The smallest unit of a Kubernetes-managed containerized application. A pod consists of one or more containers

- Deployments: The operational unit that provides granular management of a running application.

- Projects: A Kubernetes namespace with additional annotations that provide multitenancy scoping for applications.

- Routes: Networking configuration to expose your applications and services to resources outside the cluster.

- Operators: Packaged Kubernetes applications that extend cluster functions.

These concepts are covered in more detail throughout the course. You can find these concepts throughout the web console as you explore the features of an OpenShift cluster from the graphical environment.

Guided Exercise: Navigate the OpenShift Web Console

You should be able to:

- Explore the features and components of Red Hat OpenShift by using the web console.

- Create a sample application by using the Developer perspective in the web console.

- Switch to the Administrator perspective and examine the resources that are created for the sample application.

- Use the web console to describe the cluster nodes, networking, storage, and authentication.

- View the default cluster operators, pods, deployments, and services.

Chapter Summary

In this chapter, you learned:

- Containers are an isolated application runtime created with very little overhead.

- A container image packages an application with all of its dependencies, making it easier to run the application in different environments.

- Applications such as Podman create containers using features of the standard Linux kernel.

- Container image registries are the preferred mechanism for distributing container images to multiple users and hosts.

- OpenShift orchestrates applications composed of multiple containers using Kubernetes.

- Kubernetes manages load balancing, high availability, and persistent storage for containerized applications.

- OpenShift adds to Kubernetes multitenancy, security, ease of use, and continuous integration and continuous development features.

- OpenShift routes enable external access to containerized applications in a manageable way.