

Deploy to Kubernetes with Load Balancing

Overview

This type of deployment is typically used for production: tools such as failover and horizontal scaling can be used with this environment.

Adding Traefik to the environment provides a reverse proxy, load balancing and sticky sessions.

Remember, Yellowfin cluster deployments require a load balancer or a reverse proxy so that sticky sessions are available, but single Yellowfin instance deployments don't require load balancing nor a reverse proxy, as they are already stand-alone.

Choose your preferred deployment from the table below, then follow the instructions.

Deployment	Image	Description
Yellowfin sandbox	All-In-One	A self-contained instance of Yellowfin. This is the simplest type of deployment. All content will be lost when the container is destroyed.
Yellowfin single instance	App-Only	A single instance with a separate database, so data stored in the database will not be lost when the container is destroyed.
Yellowfin multiple discrete instances	App-Only	Multiple instances each with their own dedicated database. This could be used to stage a development environment and a production environment during a proof of concept.
Yellowfin cluster	App-Only	Multiple instances sharing a single database to form a Yellowfin cluster. This could be used to stage a clustered environment during a proof of concept.

[top](#)

Install a load balancer/reverse proxy

Before installing Yellowfin, install your preferred load balancer or reverse proxy server. If you don't have a preference, we have provided some instructions below to install Traefik.

Install Traefik using Helm

Traefik works as a reverse proxy, load balancer and sticky sessions manager.

To install Traefik on Kubernetes, we recommend using the Kubernetes package manager, Helm. This will help to get Traefik up and running on your cluster. Please note that these are third-party tools, so the steps below are provided as a guide only. For further details of each product, please visit their websites.

1. Start Kubernetes in your preferred terminal
2. Type the following command to check that your terminal has a connection to the Kubernetes Cluster that you wish to manage:

```
kubect1 get svc
```

3. If you have multiple Kubernetes clusters and you're in the wrong one, see the Kubernetes guide to switch clusters:

<https://kubernetes.io/docs/tasks/access-application-cluster/configure-access-multiple-clusters/>

4. Cut and paste the following commands into your terminal to install Helm3

```
curl https://raw.githubusercontent.com/helm/helm/master/scripts/get-helm-3 > get_helm.sh
chmod 700 get_helm.sh
./get_helm.sh
```

5. Cut and paste the following commands into your terminal to add Traefik's Helm chart repository:

```
helm repo add traefik https://containous.github.io/traefik-helm-chart
helm repo update
```

6. Cut and paste the following command into your terminal to install Traefik on your Kubernetes cluster:

```
helm install traefik traefik/traefik
```

Helm will now set up Traefik to be an available Ingress type into your Kubernetes cluster. Once complete, a Traefik service will be running in the Kubernetes cluster, using the Kubernetes service type "LoadBalancer".

[top](#)

Section navigation

Current topic - Install in a Container

The page is part of the [Install in a Container](#) topic contains the following pages, split by Docker and Kubernetes:

Deploy to Kubernetes with Load Balancing

- [Deploy to Docker without Swarm](#)
 - [Sandbox Instance with All-In-One Image](#)
 - [Single Instance with App-Only Image](#)
 - [Multiple Discrete Instances with App-Only Image](#)
 - [A Cluster with App-Only Image](#)
- [Deploy to Docker with Swarm](#)
 - [Sandbox instance with All-In-One Image - Swarm](#)
 - [Single Instance with App-Only Image - Swarm](#)
 - [Multiple Discrete Instances with App-Only Image - Swarm](#)
 - [A Cluster with App-Only Image - Swarm](#)

Kubernetes

- [Deploy to Kubernetes without load balancing](#)
 - [Sandbox Instance with All-In-One Image - no Load Balancer](#)
 - [Multiple Discrete Instances with App-Only Image - no Load Balancer](#)
- [Deploy to Kubernetes with Load Balancing](#)
 - [Single Instance with App-Only Image and Load Balancer](#)
 - [A Cluster with App-Only Image and Load Balancer](#)

This page is part of the [Install And Deploy Yellowfin](#) section of the wiki, which has these topics:

Install on Premises

Deploy to Kubernetes with Load Balancing

- [Installation Steps](#)

Install in the Cloud

Install in the Cloud

- [Yellowfin for AWS](#)
- [Yellowfin for Azure](#)
- [Yellowfin for Google Cloud Platform](#)

Install in a container

Install in a Container

- [Docker](#)
- [Kubernetes](#)

- [Upgrading Yellowfin Container Deployment](#)

Deploy Yellowfin

[Deploy Yellowfin](#)

- [Logs and Logging](#)
- [Yellowfin Directory Structure](#)
- [User Welcome](#)

Advanced Deployments

[Advanced Deployments](#)

- [Clustering Guide](#)
- [Yellowfin Server Specification](#)
- [Automate Yellowfin Deployment on Linux](#)
- [SAML Bridge](#)
- [Standalone Configuration Tools](#)

[top](#)
