Step by Step JupyterHub Setup

This setup guide assumes that Kubernetes is already setup and configured on the system. Kubernetes setup guide can be found here.

1. Make Helm aware of the JupyterHub Helm chart repository so you can install the JupyterHub chart from it without having to use a long URL name.

```
microk8s helm repo add jupyterhub https://hub.jupyter.org/helm-chart/
microk8s helm repo update
```

This should show output like:

```
Hang tight while we grab the latest from your chart repositories...
...Skip local chart repository
...Successfully got an update from the "stable" chart repository
...Successfully got an update from the "jupyterhub" chart repository
Update Complete. ❈ Happy Helming!❈
```

2. Now install the chart configured by your `config.yaml` by running this command from the directory that contains your `config.yaml`:

```
helm upgrade --cleanup-on-fail \
--install <helm-release-name> jupyterhub/jupyterhub \
--namespace <k8s-namespace> \
--create-namespace \
--version=<chart-version> \
--values config.yaml
```

where:

- `<helm-release-name>` refers to a Helm release name, an identifier used to differentiate chart installations. You need it when you are changing or deleting the configuration of this chart installation. If your Kubernetes cluster will contain multiple JupyterHubs make sure to differentiate them. You can list your Helm releases with `helm list`.
- `<k8s-namespace>` refers to a Kubernetes namespace, an identifier used to group Kubernetes resources, in this case all Kubernetes resources associated with the JupyterHub chart. You'll need the namespace identifier for performing any commands with `kubectl`.
- This step may take a moment, during which time there will be no output to your terminal. JupyterHub is being installed in the background.
- If you get a `release named <helm-release-name> already exists` error, then you should delete the release by running `helm delete <helm-release-name>`. Then reinstall by repeating this step. If it persists, also do `kubectl delete namespace <k8s-namespace>` and try again.
- In general, if something goes wrong with the install step, delete the Helm release by running `helm delete <helm-release-name>` before re-running the install command.
- If you're pulling from a large Docker image you may get a `Error: timed out waiting for the condition` error, add a —`timeout=<number-of-minutes>m` parameter to the `helm` command.
- The —`version` parameter corresponds to the version of the Helm chart, not the version of

JupyterHub. Each version of the JupyterHub Helm chart is paired with a specific version of JupyterHub. E.g., `0.11.1` of the Helm chart runs JupyterHub `1.3.0`. For a list of which JupyterHub version is installed in each version of the JupyterHub Helm Chart, see the Helm Chart repository.

From:
https://wiki.eolab.de/ - **HSRW EOLab Wiki**

Permanent link:
https://wiki.eolab.de/doku.php?id=eolab:crunchy_cloud:jupyterhub_setup:start

Last update: **2024/12/18 11:27**