f8 Installation Guide

The following sections will walk you through installing f8 on a supported platform. Each command is meant to be run from a terminal window that has the documented prerequisites installed as well as access configured to the provider on which you want to install f8. This installer can be run from **MacOS**, **Linux**, or **Windows 10** (via WSL).

Prerequisites

General

The following prequisites are needed before you can install f8 on any platform. Click on any for a link with more information on how to install it.

Note that the installation of these may be deferred until after platform-specific prequisites are installed (see the following section).

- Windows Subsystem Linux (WSL) 2 **Windows-only**
    - This only applies if you are installing from Windows.
    - The remaining steps in this guide will need to be run from a terminal in the Linux distribution you install (Ubuntu is recommended) when following the WSL installation guide in the link above.
    - Note that you may need to restart you computer during installation of this prerequisite.
- Python 2
    - Check if you already have it by running the following command:

      ```
      python2.7 -V
      ```

    - Also confirm `pip`, the Python package manager is installed by running the following command:

      ```
      python2.7 -m pip -V
      ```

        - If this fails, run the following commands to install `pip`:

          ```
          curl -o get-pip.py https://bootstrap.pypa.io/pip/2.7/get-pip.py
          python2.7 get-pip.py
          ```

    - *We are aware that Python 2 has reached end-of-life status, and are actively working on moving f8 to Python 3.*
- Docker
    - For **Windows**, use this link.
    - For **Linux Ubuntu** (e.g. if installing f8 on Raspberry Pi), the recommended method of installation is running the following:

        -
          ```
          sudo apt-get install -y docker.io
          ```

    - Confirm it is installed/running with the following command (prepending `sudo` if needed):

      ```
      docker images
      ```

Platform

This section lists prerequisites specific to the platform on which f8 is being installed. Choose one of the following.

Google Cloud Platform (GCP)

This is the recommended provider, particularly if you do not already have a Kubernetes cluster to install f8 on, as the installer can automatically create one for you on GCP. It also supports fully automated installation if you supply the `--auto` option to the installation command (assuming the prerequisites in this section are met). GCP also provides $300 of free trial credit, which is enough to run f8 for around one month.

- gcloud (Google Cloud SDK)
  - Choose instructions for the appropriate platform when installing from the above link. For **Windows**, be sure to follow the instructions for the Linux distribution you are running.
  - Your gcloud CLI must be logged in to an active account. Confirm this by running the following command to check for an active account:

    ```
    gcloud auth list
    ```

  - If you require **sudo** to run docker commands, you must additionally log in to gcloud with:

    ```
    sudo gcloud auth login
    ```

  - If you have not set up your account yet with GCP, you will need to navigate to https://console.cloud.google.com and accept the terms and conditions.
  - During installation, you may also be asked to specify a GCP project which has billing enabled so you can create resources in it. See the following section for an estimate on the cost of any resources that may be created during installation.

Amazon Web Services (AWS)

If installing f8 on AWS, you will need to have a Kubernetes cluster already running that you can specify when prompted for it in the installer. See here for documentation on how to create one.

Raspberry Pi

This requires a Raspberry Pi with at least 4GB RAM and 64GB storage, running Ubuntu Server 20 LTS, with the following additional prequisites met.

- cgroups must be enabled. This can be done with the following:
  - Edit the file `/boot/firmware/cmdline.txt` (using `sudo`, e.g. `sudo vim /boot/firmware/cmdline.txt`), and append the below to the end of the existing line:

    -
      ```
      cgroup_enable=cpuset cgroup_enable=memory cgroup_memory=1
      ```
  - Reboot your Pi with `sudo reboot`.
  - Confirm the update was successful by running `cat /proc/cmdline` and locating the text you added above.
- The Pi must also be connected to the Internet. The following are instructions for setting this up, otherwise alertnate guides for this can be found online.
  - Edit the file `/etc/netplan/50-cloud-init.yaml` (using `sudo`) and add the below directly under the `network:` section:

        -
          ```
          wifis:
              wlan0:
                  dhcp4: true
                  optional: true
                  access-points:
                      "YOUR-WIFI-NETWORK-NAME":
                          password: "YOUR-WIFI-PASSWORD"
          ```
    - Run `sudo netplan apply`, wait a minute for it to take effect, then confirm you can successfully reach the Internet, e.g. confirm the following works: `curl www.google.com`.
    - Run the following command to obtain the IP address of the Pi, for use in the following section:

        -
          ```
          hostname -I | cut -d ' ' -f 1
          ```
- Your f8 license file should be copied to the Pi. This can be done by running the below from the **computer on which you downloaded the license**:

    -
      ```
      scp <your-downloads-folder>/f8-license.json ubuntu@<Pi-IP-address>:
      ```
    - If you plan to integrate f8 with GitHub, Slack or other external applications that require the copy/pasting of keys, log in to the Pi from the same computer and run the remaining commands in this guide from there:

        -
          ```
          ssh ubuntu@<Pi-IP-address>
          ```
- The following additional commands must be run:

    -
      ```
      sudo apt-get install -y python-subprocess32
      ```
        - Note that if you see an error describing that an interfering process is using the necessary lock, please wait 10 minutes and try again.
- If you deferred installation of the general prerequisites (e.g. until an Internet connection was established), please go back and ensure those are completed as well.

Registry Provider

This section lists prerequisites specific to the container registry provider you want f8 to integrate with. Choose one of the following.

Google Container Registry (GCR)

GCR is automatically configured for you if you are letting the installer create a Kubernetes cluster for you.

Otherwise, you will be asked for a service account key (JSON) file that has access to push images to the GCR location you specify during installation.

DockerHub

- Create the following repositories in your DockerHub organization:
    - `<dockerhub-org>/f8-ui`
    - `<dockerhub-org>/f8-api`

    - `<dockerhub-org>/f8-auth-api`
    - `<dockerhub-org>/f8-func`
    - `<dockerhub-org>/f8-perf`
    - `<dockerhub-org>/f8-jenkins`
    - `<dockerhub-org>/f8-jenkins-slave`
- During installation, you will also be asked to provide credentials for a user that can be used by f8 to push images to these repositories.

AWS Elastic Container Registry (ECR)

- Create the following repositories in your ECR registry:
    - `<ecr-registry>/f8/ui`
    - `<ecr-registry>/f8/api`
    - `<ecr-registry>/f8/auth-api`
    - `<ecr-registry>/f8/func`
    - `<ecr-registry>/f8/perf`
    - `<ecr-registry>/f8/jenkins`
    - `<ecr-registry>/f8/jenkins-slave`
- During installation, you will also be asked to provide credentials for a user that can be used by f8 to push images to these repositories.

Local (for Raspberry Pi)

When installing on a Raspberry Pi, a local registry is automatically configured for you by default.

Installation

1. Install the f8 CLI by opening a terminal to the installer directory (where this guide lives) and running the following.

    -
      ```
      ./install-cli.sh
      ```
        - If you see permission errors, try any suggested workarounds presented in the output and proceed to the next step if they succeed.
2. Run the f8 installer with the following command, which will begin a series of prompts that will guide you through the installation of your f8 instance.

    -
      ```
      f8 install --from us.gcr.io/f8-dev1-xyz
      ```
        - For **Raspberry Pi** installation, simply run `f8 install`.
    - The installer will try its best to report details/remediation on any errors it encounters, but for cases where it cannot, see the **Troubleshooting** section for more possible information.
    - The `--from` option informs the installer to pull container images for f8 from the publicly hosted container registry specified above.
    - The `--from-tag` option can additionally be provided to specify the version of the f8 container images to install (defaults to "latest").
    - If you are installing on GCP, you can append the `--auto` option to the above command to assume all possible defaults. Also note the following if you choose to supply this option:
        - For GCP installation, A GKE cluster will automatically be created for you in your GCP account, see the following step for cost estimates.
        - The admin username/password of the f8 installation will be set to `admin/admin`.
        - External integrations with GitHub, Slack, etc. will be skipped during installation.

3. Follow the prompts presented by the installer.
    - The installer will first ask you to provide the path to your f8 license, which you should have received via email if you signed up for a free trial here.
    - It will then ask you to specify a Kubernetes cluster on which to install f8, and can automatically create one for you (on GCP or Raspberry Pi) if you don't already have one.
        - Note that if a Kubernetes cluster is created for you on GCP, **you will be charged by your cloud provider** for the resources this cluster consumes while it's running. With default settings for cluster creation, this will be somewhere near **$0.50/hour**. To delete the cluster (and stop being billed), see the following section on uninstallation.
    - It will also ask you for core configuration information for f8 like the DNS name you want to use to access the UI (e.g. `f8.your-company.com`) and SSL/TLS certificates, as well as credentials to integrate with your GitHub, Slack, and other accounts to provide a more integrated development experience.
4. Open your browser to the URL provided by the installer to access f8.
    - Once you log in, you may find additional documentation on using f8 in the "Docs" page.
    - Feel free to also reach out to support@aramse.io for more help!

Troubleshooting

- When running the installer, if you see an error dialog complaining about an "**unidentified developer**", follow the instructions here to make an exception for the file mentioned in the dialog, and then re-run the installer.
- Upon login to the f8 instance, if you see any errors around "Error connecting to the f8 server", it may be due to temporary downtime in the underlying Kubernetes cluster which can be caused by the bulk deploy of f8 onto it immediately following its creation. This often lasts **5-10 minutes**, after which you should be able to reload the page. A task is in progress with the f8 dev team to automate the reload of this page.

Uninstallation

f8 runs entirely in the Kubernetes cluster on which it is installed.

- If your platform is GCP and you had the installer create the cluster for you, you may delete it by navigating to the GKE console in GCP and deleting the cluster that was created by the installer.
- For Raspberry Pi installations, you can remove the Kubernetes cluster by running `sudo snap remove microk8s`.
- Otherwise if you installed f8 on a previously exisiting Kubernetes cluster, you may run the following command to remove f8 from it:

```
kubectl delete ns f8-prod f8-secrets ingress-nginx-f8
```