OpenFaaS cluster + dockerhub use

1 - Setting the Master Node:

```
barbara@barbara-Inspiron-15-3511:~$ ssh barbara@192.168.3.20
The authenticity of host '192.168.3.20 (192.168.3.20)' can't be established.
ED25519 key fingerprint is SHA256:Z1dMTgbbGOOzDzeko7XefN2S1+T6/fNVJk5aHa/r0BU.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.3.20' (ED25519) to the list of known hosts.
barbara@192.168.3.20's password:
Welcome to Ubuntu 22.04.1 LTS (GNU/Linux 5.15.0-52-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

  System information as of Thu Oct 27 10:08:04 PM UTC 2022

  System load:  0.0               Processes:               113
  Usage of /:   46.9% of 9.74GB   Users logged in:         0
  Memory usage: 11%               IPv4 address for enp0s3: 192.168.3.20
  Swap usage:   0%                IPv4 address for enp0s8: 10.0.2.5
```

Installing k3s:

```
barbara@kube-master:~$ curl -sfL https://get.k3s.io | sh -
[sudo] password for barbara:
[INFO]  Finding release for channel stable
[INFO]  Using v1.25.3+k3s1 as release
[INFO]  Downloading hash https://github.com/k3s-io/k3s/releases/download/v1.25.3+k3s1/sha256sum-amd64.txt
[INFO]  Downloading binary https://github.com/k3s-io/k3s/releases/download/v1.25.3+k3s1/k3s
[INFO]  Verifying binary download
[INFO]  Installing k3s to /usr/local/bin/k3s
[INFO]  Skipping installation of SELinux RPM
[INFO]  Creating /usr/local/bin/kubectl symlink to k3s
[INFO]  Creating /usr/local/bin/crictl symlink to k3s
[INFO]  Creating /usr/local/bin/ctr symlink to k3s
[INFO]  Creating killall script /usr/local/bin/k3s-killall.sh
[INFO]  Creating uninstall script /usr/local/bin/k3s-uninstall.sh
[INFO]  env: Creating environment file /etc/systemd/system/k3s.service.env
[INFO]  systemd: Creating service file /etc/systemd/system/k3s.service
[INFO]  systemd: Enabling k3s unit
Created symlink /etc/systemd/system/multi-user.target.wants/k3s.service → /etc/systemd/system/k3s.service.
[INFO]  systemd: Starting k3s
```

Verifying status

```
barbara@kube-master: $ sudo systemctl status k3s
● k3s.service - Lightweight Kubernetes
     Loaded: loaded (/etc/systemd/system/k3s.service; enabled; vendor preset: enabled)
     Active: active (running) since Thu 2022-10-27 22:09:17 UTC; 1min 32s ago
       Docs: https://k3s.io
    Process: 1150 ExecStartPre=/bin/sh -xc ! /usr/bin/systemctl is-enabled --quiet nm-cloud-setup.service (code=exited, status=0/SUCCESS)
    Process: 1152 ExecStartPre=/sbin/modprobe br_netfilter (code=exited, status=0/SUCCESS)
    Process: 1156 ExecStartPre=/sbin/modprobe overlay (code=exited, status=0/SUCCESS)
   Main PID: 1157 (k3s-server)
      Tasks: 103
     Memory: 1.3G
        CPU: 27.682s
     CGroup: /system.slice/k3s.service
             ├─1157 "/usr/local/bin/k3s server"
             ├─1172 containerd -c /var/lib/rancher/k3s/agent/etc/containerd/config.toml -a /run/k3s/containerd/containerd.sock --state /run/k3s/containerd --root /var/lib/rancher/k3s/
agent/containerd
             ├─1866 /var/lib/rancher/k3s/data/2ef87ff954adbb390309ce4dc07500f29c319f84feec1719bfb5059c8808ec6a/bin/containerd-shim-runc-v2 -namespace k8s.io -id 0832234eda457ad714dc18
a849ae47f26687e1d1e8
             ├─1905 /var/lib/rancher/k3s/data/2ef87ff954adbb390309ce4dc07500f29c319f84feec1719bfb5059c8808ec6a/bin/containerd-shim-runc-v2 -namespace k8s.io -id b616dbf634e48ccb162aaa
bf15f76bfa7293dd01f8
             ├─1931 /var/lib/rancher/k3s/data/2ef87ff954adbb390309ce4dc07500f29c319f84feec1719bfb5059c8808ec6a/bin/containerd-shim-runc-v2 -namespace k8s.io -id 960e9f4bfda169491cd869
0ccc57036bc6b066c03e
             ├─2937 /var/lib/rancher/k3s/data/2ef87ff954adbb390309ce4dc07500f29c319f84feec1719bfb5059c8808ec6a/bin/containerd-shim-runc-v2 -namespace k8s.io -id c13bc081f8d8e211617939
6bd8f72e485f71c376a8
             └─3014 /var/lib/rancher/k3s/data/2ef87ff954adbb390309ce4dc07500f29c319f84feec1719bfb5059c8808ec6a/bin/containerd-shim-runc-v2 -namespace k8s.io -id 4c767f926b732f512ca70b
2844061e272922b69d6e
```

Getting the nodes:

```
barbara@kube-master:~$ sudo kubectl get nodes
NAME          STATUS   ROLES                  AGE     VERSION
kube-master   Ready    control-plane,master   5m28s   v1.25.3+k3s1
barbara@kube-master:~$
```

Configuration Kubeconfig file. If we have problems when installing helm, for example, we need to take the data of this file and export to the desired path, configuring the correct kubeconfig path:


```
barbara@kube-master:~$ sudo cat /etc/rancher/k3s/k3s.yaml
apiVersion: v1
clusters:
- cluster:
    certificate-authority-data: LS0tLS1CRUdJTiBDRVJUSUZJQ0FURS0tLS0tCk1JSUJlRENDQVIyZ0F3SUJBZ0lCQURBS0JnZ3Foa2pPUFFRREFqQWpNU0V3SHdZRFZRUUREQmhyTTNNdGMyVnkKZG1WeUxXTmhRREUyTmpZNU1EZzF
OVE13SGhjTk1qSXhNREkzTWpJd09URXpXaGNOTXpJeE1ESTBNakl3T1RFegpXakFqTVNFd0h3WURWUVFERE Jock0zTXRjMlZ5ZG1WeUxXTmhRREUyTmpZNU1EZzFOVE13V1RBVEJnY3Foa2pPClBRSUJCZ2dxaGtqT1BRTUJCd05DQUFTWkNEeG
ZoVTA1N2ZnMU5KK1FPU0JRTVBodTZpd0F3c3I4WWFSVFhtN0QKekZzK3ZacS9oYXlmQ1JtdEpoYTB4OUtWYWJ6OXZiUVhZVjBBcmUveWFHMGNvMEl3UURBT0JnTlZIUThCQWY4RQpCQU1DQXFRd0R3WURWUjBUQVFIL0JBVXdBd0VCL3pBZEJnT
lZIUTRFRmdRVWhlT2l4VjVWTmxYSkZZVFJLTnVPCjI5bnpXQTh3Q2dZSUtvWkl6ajBFQXdJRFNRQXdSZ0loQU0yc3BENXgrc1dHTzVwanJvRTY3blRyNFhrQkJPSFUKQUJqVnJEUTFpVTBLQWlFQTJFVENJNEh3Q2JNRVd2elhoRXA1MTkrUGxr
c25lL1FXbDVQMG9wL1ZnWGs9Ci0tLS0tRU5EIENFUlRJRklDQVRFLS0tLS0K
    server: https://127.0.0.1:6443
  name: default
contexts:
- context:
    cluster: default
    user: default
  name: default
current-context: default
kind: Config
preferences: {}
users:
- name: default
  user:
    client-certificate-data: LS0tLS1CRUdJTiBDRVJUSUZJQ0FURS0tLS0tCk1JSUJrakNDQVRlZ0F3SUJBZ0lJYWZGWCtKbm9HY0l3Q2dZSUtvWkl6ajBFQXdJd0l6RWhNQjhHQTFVRUF3d1kKYXpOekxXTnNhV1Z1ZEMxallVQXhOal
kyT1RBNE5UVXpNQjRYRFRJeU1UQXlOekl5TURreE0xb1hEVEl6TVRBeQpOekl5TURreE0xb3dNREVYTUJVR0ExVUVDaE1PYzNsemRHVnRPbTFoYzNSbGNuTXhGVEFUQmdOVkJBTVRESE41CmMzUmxiVHBoWkcxcGJqQlpNQk1HQnlxR1NNNDlBZ
0VHQ0NxR1NNNDlBd0VIQTBJQUJLUWZGY1c2dWJaaERRMWwKY3JYczd6NXU0eFo1blliWC9YU294VDg1TUhQWnYwUURWQjdpZEdySGM3WHZ1ajdvc2dqTTJhNHd4cGNvQ1BSbwpwN2hTeG95alNEQkdNQTRHQTFVZER3RUIvd1FFQXdJRm9EQVRC
Z05WSFNVRUREQUtCZ2dyQmdFRkJRY0RBakFmCkJnTlZIU01FR0RBV2dCU3ZRSU5keXNzbHZiQlkxRUF6b29VUThlTWpaakFLQmdncWhrak9QUVFEQWdOSkFEQkcKQWlFQW1VV09QbmMxc2JwaSswL0Y5TmlJSFpaWUhXQStXYUtpQ3ZzOGpKNEh
0aWtDSVFDTFZBMWhaMTMvaXJLUWpXTnFPSXdoNThQTGZsR3o5ZFhIVGhmZThObyswK0E9PQotLS0tLUVORCBDRVJUSUZJQ0FURS0tLS0tCi0tLS0tQkVHSU4gQ0VSVElGSUNBVEUtLS0tLQpNSUlCZURDQ0FSMmdBd0lCQWdJQkFEQUtCZ2dxaG
tqT1BRUURBakFqTVNFd0h3WURWUVFERE Jock0zTXRZMnhwClpXNTBMV05oUURFMk5qWTVNRGcxTlRNd0hoY05Nakl4TURJM01qSXdPVEV6V2hjTk16SXhNREkwTWpJd09URXoKV2pBak1TRXdId1lEVlFRRERCaHJNM010WTJ4cFpXNTBMV05oU
URFMk5qWTVNRGcxTlRNd1dUQVRCZ2NxaGtqTwpQUUlCQmdncWhrak9QUU1CQndOQ0FBUTBUY1hrZXRSK05HZGRFMjlaL2ZjZEZhMmVaUEs2V3dUT0s2cEFLNzFuCmFkdHVWYjM0Y1BzNm1zbG0rRzVGK0VBUTRyVDY5RGZ2VmIyRldVeHVsdUxQ
bzBJd1FEQU9CZ05WSFE4QkFmOEUKQkFNQ0FxUXdEd1lEVlIwVEFRSC9CQVV3QXdFQi96QWRCZ05WSFE0RUZnUVVyMENEWGNyTEpiMndXTlJBTTZLRgpFUEhqSTJZd0NnWUlLb1pJemowRUF3SURTUUF3UmdJaEFMOHFXSVhKN0J3R2k1SXVJSi9
nRFNMU1RwUlRtTnJmCnlmNjN1UGJyOFdJU0FpRUEzd3ovR1ZHc2VWUW50Q3BRR2RiN1JlaHJNZUpYaEoycVh0ZWdzOERDcUxZPQotLS0tLUVORCBDRVJUSUZJQ0FURS0tLS0tCg==
    client-key-data: LS0tLS1CRUdJTiBFQyBQUklWQVRFIEtFWS0tLS0tCk1IY0NBUUVFSUJmL1ZTREtVYWlST0hoOWdRWVg4cFJZTDFpWWMxNjhtOUptTDUxQ1p1TzNvQW9HQ0NxR1NNNDkKQXdFSG9VUURRZ0FFcEI4VnhicTV0bUVORF
dWeXRlenZQbTdqRm5tZGh0ZjlkS2pGUHprd2M5bS9SQU5VSHVKMAphc2R6dGUrNlB1aXlDTXpacmpER2x5Z0k5R2ludUZMR2pBPT0KLS0tLS1FTkQgRUMgUFJJVkFURSBLRVktLS0tLQo=
barbara@kube-master:~$
```


```
barbara@kube-master:~$ sudo kubectl get all --all-namespaces
NAMESPACE     NAME                                          READY   STATUS      RESTARTS   AGE
kube-system   pod/local-path-provisioner-5b5579c644-d4tfm   1/1     Running     0          27m
kube-system   pod/coredns-75fc8f8fff-dp5h8                  1/1     Running     0          27m
kube-system   pod/helm-install-traefik-crd-krgh5            0/1     Completed   0          27m
kube-system   pod/svclb-traefik-b650ac6a-5jntr              2/2     Running     0          26m
kube-system   pod/helm-install-traefik-rb5d4                0/1     Completed   1          27m
kube-system   pod/metrics-server-5c8978b444-pvxs6           1/1     Running     0          27m
kube-system   pod/traefik-9c6dc6686-645sh                   1/1     Running     0          26m

NAMESPACE     NAME                     TYPE           CLUSTER-IP     EXTERNAL-IP   PORT(S)                      AGE
default       service/kubernetes       ClusterIP      10.43.0.1      <none>        443/TCP                      27m
kube-system   service/kube-dns         ClusterIP      10.43.0.10     <none>        53/UDP,53/TCP,9153/TCP       27m
kube-system   service/metrics-server   ClusterIP      10.43.248.66   <none>        443/TCP                      27m
kube-system   service/traefik          LoadBalancer   10.43.33.208   10.0.2.5      80:30466/TCP,443:30790/TCP   26m

NAMESPACE     NAME                                    DESIRED   CURRENT   READY   UP-TO-DATE   AVAILABLE   NODE SELECTOR   AGE
kube-system   daemonset.apps/svclb-traefik-b650ac6a   1         1         1       1            1           <none>          26m

NAMESPACE     NAME                                     READY   UP-TO-DATE   AVAILABLE   AGE
kube-system   deployment.apps/local-path-provisioner   1/1     1            1           27m
kube-system   deployment.apps/coredns                  1/1     1            1           27m
kube-system   deployment.apps/metrics-server           1/1     1            1           27m
kube-system   deployment.apps/traefik                  1/1     1            1           26m

NAMESPACE     NAME                                                DESIRED   CURRENT   READY   AGE
kube-system   replicaset.apps/local-path-provisioner-5b5579c644   1         1         1       27m
kube-system   replicaset.apps/coredns-75fc8f8fff                  1         1         1       27m
kube-system   replicaset.apps/metrics-server-5c8978b444           1         1         1       27m
kube-system   replicaset.apps/traefik-9c6dc6686                   1         1         1       26m

NAMESPACE     NAME                                 COMPLETIONS   DURATION   AGE
kube-system   job.batch/helm-install-traefik-crd   1/1           27s        27m
kube-system   job.batch/helm-install-traefik       1/1           31s        27m
```

```
barbara@kube-master:~$ git clone https://github.com/alexellis/iot-sensors-mqtt-openfaas.git
Cloning into 'iot-sensors-mqtt-openfaas'...
remote: Enumerating objects: 29, done.
remote: Counting objects: 100% (1/1), done.
remote: Total 29 (delta 0), reused 0 (delta 0), pack-reused 28
Receiving objects: 100% (29/29), 339.68 KiB | 647.00 KiB/s, done.
Resolving deltas: 100% (6/6), done.
```

Installing docker
https://docs.docker.com/engine/install/linux-postinstall/

```
barbara@kube-master:~$ sudo apt-get install \
    ca-certificates \
    curl \
    gnupg \
    lsb-release
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
ca-certificates is already the newest version (20211016).
ca-certificates set to manually installed.
lsb-release is already the newest version (11.1.0ubuntu4).
lsb-release set to manually installed.
curl is already the newest version (7.81.0-1ubuntu1.6).
curl set to manually installed.
gnupg is already the newest version (2.2.27-3ubuntu2.1).
gnupg set to manually installed.
0 upgraded, 0 newly installed, 0 to remove and 45 not upgraded.
barbara@kube-master:~$ sudo mkdir -p /etc/apt/keyrings
barbara@kube-master:~$

curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo gpg --dearmor -o /etc/apt/keyrings/docker.gpg
barbara@kube-master:~$ echo \
  "deb [arch=$(dpkg --print-architecture) signed-by=/etc/apt/keyrings/docker.gpg] https://download.docker.com/linux/ubuntu \
  $(lsb_release -cs) stable" | sudo tee /etc/apt/sources.list.d/docker.list > /dev/null
barbara@kube-master:~$ sudo apt-get update
Hit:1 http://br.archive.ubuntu.com/ubuntu jammy InRelease
Hit:2 http://br.archive.ubuntu.com/ubuntu jammy-updates InRelease
Hit:3 http://br.archive.ubuntu.com/ubuntu jammy-backports InRelease
Hit:4 http://br.archive.ubuntu.com/ubuntu jammy-security InRelease
Get:5 https://download.docker.com/linux/ubuntu jammy InRelease [48.9 kB]
Get:6 https://download.docker.com/linux/ubuntu jammy/stable amd64 Packages [9481 B]
Fetched 58.3 kB in 1s (64.0 kB/s)
Reading package lists... Done
```

```
barbara@kube-master:~$ sudo apt-get install docker-ce docker-ce-cli containerd.io docker-compose-plugin
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following packages were automatically installed and are no longer required:
  bridge-utils dns-root-data dnsmasq-base ubuntu-fan
Use 'sudo apt autoremove' to remove them.
The following additional packages will be installed:
  docker-ce-rootless-extras docker-scan-plugin libltdl7 libslirp0
  slirp4netns
Suggested packages:
  aufs-tools cgroupfs-mount | cgroup-lite
The following packages will be REMOVED:
  containerd docker.io runc
The following NEW packages will be installed:
  containerd.io docker-ce docker-ce-cli docker-ce-rootless-extras
  docker-compose-plugin docker-scan-plugin libltdl7 libslirp0
  slirp4netns
0 upgraded, 9 newly installed, 3 to remove and 45 not upgraded.
Need to get 111 MB of archives.
After this operation, 146 MB of additional disk space will be used.
Do you want to continue? [Y/n] Y
Get:1 http://br.archive.ubuntu.com/ubuntu jammy/main amd64 libltdl7 amd64 2.4.6-15build2 [39.6 kB]
Get:2 http://br.archive.ubuntu.com/ubuntu jammy/main amd64 libslirp0 amd64 4.6.1-1build1 [61.5 kB]
Get:3 http://br.archive.ubuntu.com/ubuntu jammy/universe amd64 slirp4netns amd64 1.0.1-2 [28.2 kB]
Get:4 https://download.docker.com/linux/ubuntu jammy/stable amd64 containerd.io amd64 1.6.9-1 [27.7 MB]
Get:5 https://download.docker.com/linux/ubuntu jammy/stable amd64 docker-ce-cli amd64 5:20.10.21~3-0~ubuntu-jammy [41.5 MB]
Get:6 https://download.docker.com/linux/ubuntu jammy/stable amd64 docker-ce amd64 5:20.10.21~3-0~ubuntu-jammy [20.5 MB]
Get:7 https://download.docker.com/linux/ubuntu jammy/stable amd64 docker-ce-rootless-extras amd64 5:20.10.21~3-0~ubuntu-jammy [8389
Get:8 https://download.docker.com/linux/ubuntu jammy/stable amd64 docker-compose-plugin amd64 2.12.2~ubuntu-jammy [9566 kB]
Get:9 https://download.docker.com/linux/ubuntu jammy/stable amd64 docker-scan-plugin amd64 0.21.0~ubuntu-jammy [3622 kB]
Fetched 111 MB in 34s (3304 kB/s)
(Reading database ... 73715 files and directories currently installed.)
```

problems..

```
Setting up libslirp0:amd64 (4.6.1-1build1) ...
Setting up docker-ce-rootless-extras (5:20.10.21~3-0~ubuntu-jammy) ...
Setting up slirp4netns (1.0.1-2) ...
Setting up docker-ce (5:20.10.21~3-0~ubuntu-jammy) ...
Job for docker.service failed because the control process exited with error code.
See "systemctl status docker.service" and "journalctl -xeu docker.service" for details.
invoke-rc.d: initscript docker, action "start" failed.
● docker.service - Docker Application Container Engine
     Loaded: loaded (/lib/systemd/system/docker.service; enabled; vendor preset: enabled)
     Active: activating (auto-restart) (Result: exit-code) since Thu 2022-10-27 22:43:33 UTC; 5ms ago
TriggeredBy: ● docker.socket
       Docs: https://docs.docker.com
    Process: 6241 ExecStart=/usr/bin/dockerd -H fd:// --containerd=/run/containerd/containerd.sock (code=exited, status=1/FAILURE)
   Main PID: 6241 (code=exited, status=1/FAILURE)
        CPU: 39ms
dpkg: error processing package docker-ce (--configure):
 installed docker-ce package post-installation script subprocess returned error exit status 1
Processing triggers for man-db (2.10.2-1) ...
Processing triggers for libc-bin (2.35-0ubuntu3.1) ...
Errors were encountered while processing:
 docker-ce
needrestart is being skipped since dpkg has failed
E: Sub-process /usr/bin/dpkg returned an error code (1)
barbara@kube-master:~$ sudo docker run hello-world
docker: Cannot connect to the Docker daemon at unix:///var/run/docker.sock. Is the docker daemon running?.
See 'docker run --help'.
```

Solution - taokay

```
barbara@kube-master:~$ sudo groupadd docker
groupadd: group 'docker' already exists
barbara@kube-master:~$ sudo usermod -aG docker $USER
barbara@kube-master:~$ newgrp docker
barbara@kube-master:~$ docker run hello-world
docker: Cannot connect to the Docker daemon at unix:///var/run/docker.sock. Is the docker daemon running?.
See 'docker run --help'.
```

```
barbara@kube-master:~$ sudo reboot
Connection to 192.168.3.20 closed by remote host.
Connection to 192.168.3.20 closed.
barbara@barbara-Inspiron-15-3511:~$ ssh barbara@192.168.3.20
barbara@192.168.3.20's password:
Welcome to Ubuntu 22.04.1 LTS (GNU/Linux 5.15.0-52-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

  System information as of Thu Oct 27 10:49:27 PM UTC 2022

  System load:  0.0               Users logged in:          0
  Usage of /:   61.6% of 9.74GB   IPv4 address for docker0: 172.17.0.1
  Memory usage: 27%               IPv4 address for enp0s3:  192.168.3.20
  Swap usage:   0%                IPv4 address for enp0s8:  10.0.2.5
  Processes:    136


43 updates can be applied immediately.
To see these additional updates run: apt list --upgradable
```

```
barbara@kube-master:~$ sudo docker run hello-world
[sudo] password for barbara:
Unable to find image 'hello-world:latest' locally
latest: Pulling from library/hello-world
2db29710123e: Pull complete
Digest: sha256:e18f0a777aefabe047a671ab3ec3eed05414477c951ab1a6f352a06974245fe7
Status: Downloaded newer image for hello-world:latest

Hello from Docker!
This message shows that your installation appears to be working correctly.

To generate this message, Docker took the following steps:
 1. The Docker client contacted the Docker daemon.
 2. The Docker daemon pulled the "hello-world" image from the Docker Hub.
    (amd64)
 3. The Docker daemon created a new container from that image which runs the
    executable that produces the output you are currently reading.
 4. The Docker daemon streamed that output to the Docker client, which sent it
    to your terminal.

To try something more ambitious, you can run an Ubuntu container with:
 $ docker run -it ubuntu bash
```

Getting the token to build the 3-node-cluster

```
barbara@kube-master:~$ sudo cat /var/lib/rancher/k3s/server/node-token
K102ff9c651f474accc3e96db99a370562f58b0e417500550c198feb8f7c0473a4a::server:64f07a9817abf8aaec07458a1404e694
barbara@kube-master:~$
```

Worker1:

```
barbara@barbara-Inspiron-15-3511:~$ ssh barbara@192.168.3.21
The authenticity of host '192.168.3.21 (192.168.3.21)' can't be established.
ED25519 key fingerprint is SHA256:XSGPfaZGIXu0EfoLIu+bhQSYhTQdHvcqT2p9vsBE/LE.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.3.21' (ED25519) to the list of known hosts.
barbara@192.168.3.21's password:
Welcome to Ubuntu 22.04.1 LTS (GNU/Linux 5.15.0-52-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

  System information as of Thu Oct 27 10:59:16 PM UTC 2022

  System load:  0.0               Processes:               109
  Usage of /:   46.9% of 9.74GB   Users logged in:         1
  Memory usage: 11%               IPv4 address for enp0s3: 192.168.3.21
  Swap usage:   0%                IPv4 address for enp0s8: 10.0.2.6
```

```
barbara@kube-worker1:~$ curl -sfL https://get.k3s.io | K3S_URL=https://192.168.3.20:6443 K3S_TOKEN=K102ff9c65
17abf8aaec07458a1404e694 sh -
[sudo] password for barbara:
[INFO]  Finding release for channel stable
[INFO]  Using v1.25.3+k3s1 as release
[INFO]  Downloading hash https://github.com/k3s-io/k3s/releases/download/v1.25.3+k3s1/sha256sum-amd64.txt
[INFO]  Downloading binary https://github.com/k3s-io/k3s/releases/download/v1.25.3+k3s1/k3s
[INFO]  Verifying binary download
[INFO]  Installing k3s to /usr/local/bin/k3s
[INFO]  Skipping installation of SELinux RPM
[INFO]  Creating /usr/local/bin/kubectl symlink to k3s
[INFO]  Creating /usr/local/bin/crictl symlink to k3s
[INFO]  Creating /usr/local/bin/ctr symlink to k3s
[INFO]  Creating killall script /usr/local/bin/k3s-killall.sh
[INFO]  Creating uninstall script /usr/local/bin/k3s-agent-uninstall.sh
[INFO]  env: Creating environment file /etc/systemd/system/k3s-agent.service.env
[INFO]  systemd: Creating service file /etc/systemd/system/k3s-agent.service
[INFO]  systemd: Enabling k3s-agent unit
Created symlink /etc/systemd/system/multi-user.target.wants/k3s-agent.service → /etc/systemd/system/k3s-agent
[INFO]  systemd: Starting k3s-agent
barbara@kube-worker1:~$
```

Worker 2:

```
barbara@barbara-Inspiron-15-3511:~$ ssh barbara@192.168.3.22
The authenticity of host '192.168.3.22 (192.168.3.22)' can't be established.
ED25519 key fingerprint is SHA256:oVuOC2LgoP9oSpaPJslZJ6hXskdfcavUzZ7Xl4kdUT8.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.3.22' (ED25519) to the list of known hosts.
barbara@192.168.3.22's password:
Welcome to Ubuntu 22.04.1 LTS (GNU/Linux 5.15.0-52-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

  System information as of Thu Oct 27 11:07:05 PM UTC 2022

  System load:  0.0               Processes:               99
  Usage of /:   46.9% of 9.74GB   Users logged in:         1
  Memory usage: 10%               IPv4 address for enp0s3: 192.168.3.22
  Swap usage:   0%                IPv4 address for enp0s8: 10.0.2.8
```

```
barbara@kube-worker2:~$ curl -sfL https://get.k3s.io | K3S_URL=https://192.168.3.20:6443 K3S_TOKEN=K102ff9c651f474accc3e96
17abf8aaec07458a1404e694 sh -
[sudo] password for barbara:
[INFO]  Finding release for channel stable
[INFO]  Using v1.25.3+k3s1 as release
[INFO]  Downloading hash https://github.com/k3s-io/k3s/releases/download/v1.25.3+k3s1/sha256sum-amd64.txt
[INFO]  Downloading binary https://github.com/k3s-io/k3s/releases/download/v1.25.3+k3s1/k3s
[INFO]  Verifying binary download
[INFO]  Installing k3s to /usr/local/bin/k3s
[INFO]  Skipping installation of SELinux RPM
[INFO]  Creating /usr/local/bin/kubectl symlink to k3s
[INFO]  Creating /usr/local/bin/crictl symlink to k3s
[INFO]  Creating /usr/local/bin/ctr symlink to k3s
[INFO]  Creating killall script /usr/local/bin/k3s-killall.sh
[INFO]  Creating uninstall script /usr/local/bin/k3s-agent-uninstall.sh
[INFO]  env: Creating environment file /etc/systemd/system/k3s-agent.service.env
[INFO]  systemd: Creating service file /etc/systemd/system/k3s-agent.service
[INFO]  systemd: Enabling k3s-agent unit
Created symlink /etc/systemd/system/multi-user.target.wants/k3s-agent.service → /etc/systemd/system/k3s-agent.service.
[INFO]  systemd: Starting k3s-agent
barbara@kube-worker2:~$
```

Verifying the cluster creation in the master node:

```
barbara@kube-master:~$ sudo kubectl get nodes
NAME           STATUS   ROLES                  AGE    VERSION
kube-worker1   Ready    <none>                 10m    v1.25.3+k3s1
kube-worker2   Ready    <none>                 2m5s   v1.25.3+k3s1
kube-master    Ready    control-plane,master   62m    v1.25.3+k3s1
```

```
barbara@kube-master:~$ sudo kubectl get nodes
NAME           STATUS   ROLES                  AGE    VERSION
kube-worker1   Ready    <none>                 10m    v1.25.3+k3s1
kube-worker2   Ready    <none>                 2m5s   v1.25.3+k3s1
kube-master    Ready    control-plane,master   62m    v1.25.3+k3s1
barbara@kube-master:~$ kubectl label node kube-worker1 node-role.kubernetes.io/worker=worker
WARN[0000] Unable to read /etc/rancher/k3s/k3s.yaml, please start server with --write-kubeconfig-mode to modify kube config permissions
error: error loading config file "/etc/rancher/k3s/k3s.yaml": open /etc/rancher/k3s/k3s.yaml: permission denied
barbara@kube-master:~$ sudo kubectl label node kube-worker1 node-role.kubernetes.io/worker=worker
node/kube-worker1 labeled
barbara@kube-master:~$ sudo kubectl label node kube-worker2 node-role.kubernetes.io/worker=worker
node/kube-worker2 labeled
barbara@kube-master:~$ sudo kubectl get nodes
NAME           STATUS   ROLES                  AGE     VERSION
kube-master    Ready    control-plane,master   63m     v1.25.3+k3s1
kube-worker1   Ready    worker                 11m     v1.25.3+k3s1
kube-worker2   Ready    worker                 3m32s   v1.25.3+k3s1
```

Trying to fix, accessing the file
source /etc/profile

```
barbara@kube-master: ~   barbara@kube-worker1: ~   barbara@kube-worker2: ~
      PS1='# '
    else
      PS1='$ '
    fi
  fi
fi

if [ -d /etc/profile.d ]; then
  for i in /etc/profile.d/*.sh; do
    if [ -r $i ]; then
      . $i
    fi
  done
  unset i
fi
for kconfig in $HOME/.kube/config $(find $HOME/.kube/ -iname "*.config"
        do
                if [ -f "$kconfig" ];then
                        export KUBECONFIG=$KUBECONFIG:$kconfig
                fi
        done
~
~
~
~
:x!
```

Error: Kubernetes cluster unreachable: Get "http://localhost:8080/version": dial tcp 127.0.0.1:8080: connect: connection refused

What fixed the problem:

KUBECONFIG =~/.kube/config
mkdir ~/.kube 2> /dev/null
sudo k3s kubectl config view --raw > "$KUBECONFIG"
chmod 600 "$KUBECONFIG"

```
root@kube-master:/home/barbara# KUBECONFIG=~/.kube/config
root@kube-master:/home/barbara# mkdir ~/.kube 2> /dev/null
sudo k3s kubectl config view --raw > "$KUBECONFIG"
chmod 600 "$KUBECONFIG"
```

```
root@kube-master:/home/barbara# helm upgrade openfaas --install openfaas/openfaas      --namespace openfaas      --set functionNamespace=openfaas-fn      --set generateBasicAuth=true
Release "openfaas" does not exist. Installing it now.
NAME: openfaas
LAST DEPLOYED: Fri Oct 28 00:15:57 2022
NAMESPACE: openfaas
STATUS: deployed
REVISION: 1
TEST SUITE: None
NOTES:
To verify that openfaas has started, run:

  kubectl -n openfaas get deployments -l "release=openfaas, app=openfaas"
To retrieve the admin password, run:

  echo $(kubectl -n openfaas get secret basic-auth -o jsonpath="{.data.basic-auth-password}" | base64 --decode)
root@kube-master:/home/barbara#
```

Retrieving the adm password:
 echo $(kubectl -n openfaas get secret basic-auth -o jsonpath="{.data.basic-auth-password}" | base64 --decode)
CsN76EDp8FRS

```
root@kube-master:/home/barbara#  echo $(kubectl -n openfaas get secret basic-auth -o jsonpath="{.data.basic-auth-password}" | base64 --decode)
CsN76EDp8FRS
root@kube-master:/home/barbara#
```

https://artifacthub.io/packages/helm/openfaas/openfaas#deploy-openfaas-community-edition

```
root@kube-master:~# curl -sSL https://cli.openfaas.com | sudo sh
Finding latest version from GitHub
0.14.11
Downloading package https://github.com/openfaas/faas-cli/releases/download/0.14.11/faas-cli as /tmp/faas-cli
Download complete.

Running with sufficient permissions to attempt to move faas-cli to /usr/local/bin
New version of faas-cli installed to /usr/local/bin
Creating alias 'faas' for 'faas-cli'.
 ___                   _____           ____
  / _ \ _ __   ___ _ __ |  ___|_ _  __ _ / ___|
| | | | '_ \ / _ \ '_ \| |_ / _` |/ _` \___ \
| |_| | |_) |  __/ | | |  _| (_| | (_| |___) |
 \___/| .__/ \___|_| |_|_|  \__,_|\__,_|____/
      |_|

CLI:
 commit:  8820d8e4a15dab900d8a7e8fc271851ccb94012e
 version: 0.14.11
```

```
root@kube-master:~# mkdir -p ~/functions && \
  cd ~/functions
root@kube-master:~/functions# faas-cli new --lang python hello-python
2022/10/28 01:19:16 No templates found in current directory.
2022/10/28 01:19:16 Attempting to expand templates from https://github.com/openfaas/templates.git
2022/10/28 01:19:17 Fetched 17 template(s) : [csharp dockerfile go java11 java11-vert-x node node12 node12-debian node14 node16 node17 php7 php8 python python3
 python3-debian ruby] from https://github.com/openfaas/templates.git
Folder: hello-python created.
 ___                   _____           ____
/ _ \ _ __   ___ _ __ |  ___|_ _  __ _/ ___|
| | | | '_ \ / _ \ '_ \| |_ / _` |/ _` \___ \
| |_| | |_) |  __/ | | |  _| (_| | (_| |___) |
\___/| .__/ \___|_| |_|_|  \__,_|\__,_|____/
     |_|


Function created in folder: hello-python
Stack file written: hello-python.yml
root@kube-master:~/functions#
```

Accessing handler.py:

```
root@kube-master:~/functions# vi hello-python/handler.py
root@kube-master:~/functions#
```

```
def handle(req):
    print("Hello! You said: " + req)

    return req
~
~
~
~
~
~
~
```

```
root@kube-master:~/functions# faas-cli build -f ./hello-python.yml
[0] > Building hello-python.
Clearing temporary build folder: ./build/hello-python/
Preparing: ./hello-python/ build/hello-python/function
Building: hello-python:latest with python template. Please wait..
Sending build context to Docker daemon  8.192kB
Step 1/31 : FROM --platform=${TARGETPLATFORM:-linux/amd64} ghcr.io/openfaas/classic-watchdog:0.2.1 as watchdog
0.2.1: Pulling from openfaas/classic-watchdog
e1749af664ff: Pulling fs layer
e1749af664ff: Verifying Checksum
e1749af664ff: Download complete
e1749af664ff: Pull complete
Digest: sha256:640de69b1d683cbfa73fd3b2d707d33a4e4570164c9795c3be028949688e5c61
Status: Downloaded newer image for ghcr.io/openfaas/classic-watchdog:0.2.1
 ---> 021a98fdbddd
Step 2/31 : FROM --platform=${TARGETPLATFORM:-linux/amd64} python:2.7-alpine
2.7-alpine: Pulling from library/python
aad63a933944: Pulling fs layer
259d822268fb: Pulling fs layer
10ba96d218d3: Pulling fs layer
44ba9f6a4209: Pulling fs layer
44ba9f6a4209: Waiting
259d822268fb: Verifying Checksum
259d822268fb: Download complete
aad63a933944: Verifying Checksum
aad63a933944: Download complete
aad63a933944: Pull complete
259d822268fb: Pull complete
44ba9f6a4209: Verifying Checksum
44ba9f6a4209: Download complete
10ba96d218d3: Verifying Checksum
```

```
[0] < Building hello-python done in 19.35s.
[0] Worker done.

Total build time: 19.35s
```

Modifying the hello-python.yml to push to my docker hub account

```
version: 1.0
provider:
  name: openfaas
  gateway: http://127.0.0.1:8080
functions:
  hello-python:
    lang: python
    handler: ./hello-python
    image: barbaraazeitona/hello-python
~
```

```
version: 1.0
provider:
  name: openfaas
  gateway: http://127.0.0.1:8080
functions:
  hello-python:
    lang: python
    handler: ./hello-python
    image: barbaraazeitona/hello-python:latest
~
```

Building the function again, with the changes:

```
root@kube-master:~/functions# faas-cli build -f ./hello-python.yml
[0] > Building hello-python.
Clearing temporary build folder: ./build/hello-python/
Preparing: ./hello-python/ build/hello-python/function
Building: barbaraazeitona/hello-python:latest with python template.
Sending build context to Docker daemon  8.192kB
Step 1/31 : FROM --platform=${TARGETPLATFORM:-linux/amd64} ghcr.io/o
 ---> 021a98fdbddd
Step 2/31 : FROM --platform=${TARGETPLATFORM:-linux/amd64} python:2.
 ---> 8579e446340f
```

And pushing to docker:

```
root@kube-master:~/functions# faas-cli push -f ./hello-python.yml
[0] > Pushing hello-python [barbaraazeitona/hello-python:latest]
The push refers to repository [docker.io/barbaraazeitona/hello-python]
8a79c3d63bf1: Pushed
a41dd80c84eb: Pushed
2a599820c788: Pushed
1480173d845f: Pushed
2e8846ef1793: Pushed
3b6ed1a9514e: Pushed
b9fb0839538c: Pushed
14b200ea4aad: Pushed
e32536d98841: Pushed
ae81a1e7b31e: Pushed
a4ed473a10a0: Pushed
3f482af41b4c: Pushed
faf3e81c55af: Pushed
879c0d8666e3: Mounted from library/python
20a7b70bdf2f: Mounted from library/python
3fc750b41be7: Mounted from library/python
beee9f30bc1f: Mounted from library/python
latest: digest: sha256:884a02b9eb929934ead19fc12b2552cd40d01282b4532994177752a61fcca2d1 size: 4074
[0] < Pushing hello-python [barbaraazeitona/hello-python:latest] done.
[0] Worker done.
root@kube-master:~/functions#
```

It is wrong though, it gives an error of connection:

```
root@kube-master:~/functions# curl https://127.0.0.1:8080/function/hello-python -d "it's Barbara here :D"
curl: (7) Failed to connect to 127.0.0.1 port 8080 after 0 ms: Connection refused
```

```
barbara@kube-master:~$ curl -v http://localhost:8080/function/hello-python -d "it's Barbara here :D"
*   Trying 127.0.0.1:8080...
* connect to 127.0.0.1 port 8080 failed: Connection refused
*   Trying ::1:8080...
* connect to ::1 port 8080 failed: Connection refused
* Failed to connect to localhost port 8080 after 2 ms: Connection refused
* Closing connection 0
curl: (7) Failed to connect to localhost port 8080 after 2 ms: Connection refused
```

Ok, let's do it again from the creation of functions, to know what went wrong (I've already suspected that I need to do a login in openfaas before creating the function 😅)
Coming back to the snapshot in which I took the admin passport

Checking docker connection:

```
root@kube-master:/home/barbara# docker ps -l
CONTAINER ID   IMAGE         COMMAND    CREATED        STATUS                    PORTS     NAMES
fe5d7fec282a   hello-world   "/hello"   16 hours ago   Exited (0) 16 hours ago             sad_pasteur
```

Okay!
Making sure openfaas is deployed

```
root@kube-master:/home/barbara# kubectl get pods -n openfaas
NAME                                    READY   STATUS             RESTARTS             AGE
nats-7f8c77747c-ssd22                   1/1     Running            0                    14h
prometheus-6f69c7b564-cwcr6             1/1     Running            0                    14h
gateway-79458985bd-hqwzh                2/2     Running            0                    14h
queue-worker-78bd96497-qzr2s            1/1     Running            2 (14h ago)          14h
basic-auth-plugin-957dfbfb7-cfhd4       1/1     Running            0                    14h
alertmanager-646f7b5cf4-8lxtj           1/1     Running            1 (106m ago)         14h
release-name-mqtt-connector-5c6d68cf49-tlnrg   0/1     CrashLoopBackOff   42 (<invalid> ago)   13h
```

Pulling a template from the directory

```
root@kube-master:/home/barbara# faas-cli template store pull python3-flask
Fetch templates from repository: https://github.com/openfaas/python-flask-template at
2022/10/28 14:54:37 Attempting to expand templates from https://github.com/openfaas/python-flask-template
2022/10/28 14:54:38 Fetched 5 template(s) : [python27-flask python3-flask python3-flask-debian python3-ht
```

Creating a new function with this template

```
root@kube-master:/home/barbara# faas-cli new pydict --lang python3-flask-debian
Folder: pydict created.
  ___                   _____           ____
 / _ \ _ __  ___ _ __ |  ___|_ _  __ _ / ___|
| | | | '_ \/ _ \ '_ \| |_ / _` |/ _` \___ \
| |_| | |_) |  __/ | | |  _| (_| | (_| |___) |
 \___/| .__/ \___|_| |_|_|  \__,_|\__,_|____/
      |_|


Function created in folder: pydict
Stack file written: pydict.yml
```

requirements.txt:

PyDictionary==2.0.1

handler.py:

```
from PyDictionary import PyDictionary
dictionary = PyDictionary()
def handle(word):
  return dictionary.meaning(word)
```

Openfaas YAML configuration

```yaml
version: 1.0
provider:
  name: openfaas
  gateway: http://127.0.0.1:8080
functions:
  pydict:
    lang: python3-flask-debian
    handler: ./pydict
    image: localhost:5000/pydict:latest
```

we need to port-forward the OpenFaas gateway service to our localhost port

```
root@kube-master:/home/barbara# kubectl get service -n openfaas
NAME                TYPE        CLUSTER-IP      EXTERNAL-IP   PORT(S)          AGE
gateway-external    NodePort    10.43.36.111    <none>        8080:31112/TCP   14h
prometheus          ClusterIP   10.43.131.142   <none>        9090/TCP         14h
alertmanager        ClusterIP   10.43.249.206   <none>        9093/TCP         14h
basic-auth-plugin   ClusterIP   10.43.33.129    <none>        8080/TCP         14h
gateway             ClusterIP   10.43.41.130    <none>        8080/TCP         14h
gateway-provider    ClusterIP   10.43.145.109   <none>        8081/TCP         14h
nats                ClusterIP   10.43.209.1     <none>        4222/TCP         14h
root@kube-master:/home/barbara# kubectl port-forward -n openfaas svc/gateway 8080:8080
Forwarding from 127.0.0.1:8080 -> 8080
Forwarding from [::1]:8080 -> 8080
```

```
root@kube-master:/home/barbara# kubectl get service -n openfaas
NAME                TYPE        CLUSTER-IP      EXTERNAL-IP   PORT(S)          AGE
gateway-external    NodePort    10.43.36.111    <none>        8080:31112/TCP   14h
prometheus          ClusterIP   10.43.131.142   <none>        9090/TCP         14h
alertmanager        ClusterIP   10.43.249.206   <none>        9093/TCP         14h
basic-auth-plugin   ClusterIP   10.43.33.129    <none>        8080/TCP         14h
gateway             ClusterIP   10.43.41.130    <none>        8080/TCP         14h
gateway-provider    ClusterIP   10.43.145.109   <none>        8081/TCP         14h
nats                ClusterIP   10.43.209.1     <none>        4222/TCP         14h
root@kube-master:/home/barbara# kubectl port-forward -n openfaas svc/gateway 8080:8080
Forwarding from 127.0.0.1:8080 -> 8080
Forwarding from [::1]:8080 -> 8080
Handling connection for 8080
Handling connection for 8080
Handling connection for 8080
Handling connection for 8080
```

leave it running, and open another terminal

```
root@kube-master:/home/barbara# faas-cli build -f pydict.yml
[0] > Building pydict.
Clearing temporary build folder: ./build/pydict/
Preparing: ./pydict/ build/pydict/function
Building: localhost:5000/pydict:latest with python3-flask-debian template. Please wait..
Sending build context to Docker daemon  13.82kB
Step 1/36 : FROM --platform=${TARGETPLATFORM:-linux/amd64} ghcr.io/openfaas/of-watchdog:0.9.10 as watchdog
0.9.10: Pulling from openfaas/of-watchdog
c4fc21d17d12: Pulling fs layer
```

```
root@kube-master:~/functions# faas-cli push -f ./hello-python.yml
[0] > Pushing hello-python [barbaraazeitona/hello-python:latest]
The push refers to repository [docker.io/barbaraazeitona/hello-python]
8a79c3d63bf1: Layer already exists
a41dd80c84eb: Layer already exists
2a599820c788: Layer already exists
1480173d845f: Layer already exists
2e8846ef1793: Layer already exists
3b6ed1a9514e: Layer already exists
b9fb0839538c: Layer already exists
14b200ea4aad: Layer already exists
e32536d98841: Layer already exists
ae81a1e7b31e: Layer already exists
a4ed473a10a0: Layer already exists
3f482af41b4c: Layer already exists
faf3e81c55af: Layer already exists
879c0d8666e3: Layer already exists
20a7b70bdf2f: Layer already exists
3fc750b41be7: Layer already exists
beee9f30bc1f: Layer already exists
latest: digest: sha256:884a02b9eb929934ead19fc12b2552cd40d01282b4532994177752a61fcca2d1 size: 4074
[0] < Pushing hello-python [barbaraazeitona/hello-python:latest] done.
[0] Worker done.
```

Login openfaas!

```
root@kube-master:~/functions# PASSWORD=$(kubectl get secret -n openfaas basic-auth -o jsonpath="{.data.basic-auth-password}" | base64 --decode; echo)
root@kube-master:~/functions# env | grep PASSWORD
root@kube-master:~/functions# echo -n $PASSWORD | faas-cli login --username admin --password-stdin
Calling the OpenFaaS server to validate the credentials...
credentials saved for admin http://127.0.0.1:8080
```

```
root@kube-master:~/functions# faas-cli deploy -f hello-python.yml -g http://127.0.0.1:8080
Deploying: hello-python.

Deployed. 202 Accepted.
URL: http://127.0.0.1:8080/function/hello-python
```

Verifying if the function is in the openfaas pod:

```
root@kube-master:~/functions# kubectl get pods -n openfaas-fn
NAME                            READY   STATUS              RESTARTS   AGE
hello-python-bf7dbdc45-5sng7    0/1     ContainerCreating   0          10s
```

```
root@kube-master:~/functions# curl 127.0.0.1:8080/function/hello-python -d "it's Barbara here"
Hello! You said: it's Barbara here
it's Barbara here
```

NICE!

Now, let's try using MQTT

Installing mqtt

```
barbara@kube-master:~$ pip3 install paho-mqtt python-etcd
Defaulting to user installation because normal site-packages is not writeable
```

Creating mqtt conection:

```
barbara@kube-master:~/faas-netes/chart$ helm template -n openfaas --namespace openfaas mqtt-connector/ | kubectl apply -f -
error: error loading config file "/etc/rancher/k3s/k3s.yaml": open /etc/rancher/k3s/k3s.yaml: permission denied
barbara@kube-master:~/faas-netes/chart$ sudo -s
root@kube-master:/home/barbara/faas-netes/chart# helm template -n openfaas --namespace openfaas mqtt-connector/ | kubectl apply -f -
deployment.apps/release-name-mqtt-connector created
```

```
root@barbara-Inspiron-15-3511:/home/barbara# python3 send.py '{"sensor_id": 1, "temperature_c":
45}'
Connecting to test.mosquitto.org:1883
Connected with result code 0
Message "{"sensor_id": 1, "temperature_c": 45}" published to "openfaas-sensor-data"
root@barbara-Inspiron-15-3511:/home/barbara#
```