Installation

Install the Bacalhau CLI

In this tutorial, you'll learn how to install and run a job with the Bacalhau client using the Bacalhau CLI or Docker.

Step 1 - Install the Bacalhau Client

The Bacalhau client is a command-line interface (CLI) that allows you to submit jobs to the Bacalhau. The client is available for Linux, macOS, and Windows. You can also run the Bacalhau client in a Docker container.

By default, you will submit to the Bacalhau public network, but the same CLI can be configured to submit to a private Bacalhau network. For more information, please read Running Bacalhau on a Private Network.

Step 1.1 - Install the Bacalhau CLI

Linux/macOS (CLI) Windows (CLI) Docker

You can install or update the Bacalhau CLI by running the commands in a terminal. You may need sudo mode or root password to install the local Bacalhau binary to `/usr/local/bin`:

```
curl -sL https://get.bacalhau.org/install.sh | b
```

Step 1.2 - Verify the Installation

To verify installation and check the version of the client and server, use the `version` command. To run a Bacalhau client command with Docker, prefix it with `docker run ghcr.io/bacalhau-project/bacalhau:latest`.

Linux/macOS/Windows (CLI) Docker

```
bacalhau version
```

If you're wondering which server is being used, the Bacalhau Project has a demo network that's shared with the community. This network allows you to familiarize with Bacalhau's capabilities and launch jobs from your computer without maintaining a compute cluster on your own.

Step 2 - Submit a Hello World job

To submit a job in Bacalhau, we will use the `bacalhau docker run` command. The command runs a job using the Docker executor on the node. Let's take a quick look at its syntax:

```
bacalhau docker run [FLAGS] IMAGE[:TAG] [COMMAND]
```

CLI Docker

```
bacalhau docker run ubuntu echo Hello World
```

We will use the command to submit a Hello World job that runs an echo program within an Ubuntu container.

Let's take a look at the results of the command execution in the terminal:

```
Job successfully submitted. Job ID: f8e7789d-8e7
Checking job status... (Enter Ctrl+C to exit at

    Communicating with the network  ............
       Creating job for submission  ............
                   Job in progress  ............

To download the results, execute:
    bacalhau job get f8e7789d-8e76-4e6c-8e71-436

To get more details about the run, execute:
    bacalhau job describe f8e7789d-8e76-4e6c-8e7
```

After the above command is run, the job is submitted to the public network, which processes the job and Bacalhau prints out the related job id:

```
Job successfully submitted. Job ID: 9d20bbad-c3f
Checking job status...
```

The `job_id` above is shown in its full form. For convenience, you can use the shortened version, in this case: `9d20bbad`.

> While this command is designed to resemble Docker's run command which you may be familiar with, Bacalhau introduces a whole new set of flags to support its computing model.

Step 3 - Checking the State of your Jobs

After having deployed the job, we now can use the CLI for the interaction with the network. The jobs were sent to the public demo network, where it was processed and we can call the following functions. The `job_id` will differ for every submission.

Step 3.1 - Job status:

You can check the status of the job using `bacalhau job list` command adding the `--id-filter` flag and specifying your job id.

```
bacalhau job list --id-filter 9d20bbad
```

Let's take a look at the results of the command execution in the terminal:

```
CREATED   ID         JOB
15:24:31  0ed7617d   Type:"docker",Params:"map[Entr
                      l> EnvironmentVariables:[] Im
                      latest Parameters:[sh -c unam
                      o "Hello from Docker Bacalhau
                      Directory:]"
```

When it says `Completed`, that means the job is done, and we can get the results.

For a comprehensive list of flags you can pass to the list command check out the related CLI Reference page

Step 3.2 - Job information:

You can find out more information about your job by using `bacalhau job describe`.

```
bacalhau job describe 9d20bbad
```

Let's take a look at the results of the command execution in the terminal:

```
Job:
    APIVersion: V1beta2
    Metadata:
        ClientID: 0ff57b2521334a92e9ddab4b2f8202c887
        CreatedAt: "2023-12-21T15:24:31.750306239Z"
        ID: 0ed7617d-d5ff-40f7-8411-89830b3f3058
        Requester:
        RequesterNodeID: QmbxGSsM6saCTyKkiWSxhJCt6Fg
        RequesterPublicKey: CAASpgIwggEiMA0GCSqGSIb3
    Spec:
    ...
```

This outputs all information about the job, including stdout, stderr, where the job was scheduled, and so on.

Step 3.3 - Job download:

You can download your job results directly by using `bacalhau job get`.

```
bacalhau job get 9d20bbad
```

This results in

```
Fetching results of job '0ed7617d'...
Results for job '0ed7617d' have been written to...
/Users/test/job-0ed7617d
```

In the command below, we created a directory called `myfolder` and download our job output to be stored in that directory.

```
Fetching results of job '0ed7617d'...
Results for job '0ed7617d' have been written to...
/myfolder
```

While executing this command, you may encounter warnings regarding receive and send buffer sizes: `failed to sufficiently increase receive buffer size`. These warnings can arise due to limitations in the UDP buffer used by Bacalhau to process tasks. Additional information can be found in https://github.com/quic-go/quic-go/wiki/UDP-Buffer-Sizes.

After the download has finished you should see the following contents in the results directory.

```
job-0ed7617d
├── exitCode
├── outputs
├── stderr
└── stdout
```

Step 4 - Viewing your Job Output

```
cat job-9d20bbad/stdout
```

That should print out the string `Hello World`.

```
Hello world
```

With that, you have just successfully run a job on Bacalhau! 🐟

Step 5 - Where to go next?

Here are few resources that provide a deeper dive into running jobs with Bacalhau:

How Bacalhau works, Create your Private Network, Examples & Use Cases

Support

If you have questions or need support or guidance, please reach out to the Bacalhau team via Slack (**#general** channel).

Previous
How Bacalhau Works

Next
Create Network

Last updated 3 days ago