

Labdash

- [Install and Configure Labdash](#)

Install and Configure Labdash

<https://www.youtube.com/embed/CCEfU-ch9i4>

Install Updates

Update and upgrade your server's packages using the following commands:

- For Ubuntu/Debian: `sudo apt update && sudo apt upgrade -y`
- For RedHat/CentOS/Fedora/Alma/Rocky: `sudo dnf update -y`

Create a Non-Root User

Create a non-root user with superuser (sudo) privileges:

1. Add a new user using `adduser <username>`
2. Set the password for this user.
3. Enter the relevant information (optional)
4. Enter 'Y', then press Enter.
5. Add the user to the "sudo" group:
 - For Ubuntu/Debian: `usermod -aG sudo <username>`
 - For RedHat/CentOS/Fedora/Alma/Rocky: `usermod -aG wheel <username>`

Ubuntu / Debian

```
usermod -aG sudo <username>
```

RedHat / CentOS / Fedora / Alma / Rocky

```
usermod -aG wheel <username>
```

Now, you can log out of the system, and log back in as your new non-root super user.

Step 2: Install Docker and Docker Compose

Install Docker and Docker Compose on your server:

1. Install the `curl` utility:
 - For Ubuntu/Debian: `sudo apt install curl -y`
 - For RedHat/CentOS/Fedora/Alma/Rocky: `sudo dnf install curl -y`

2. Run the command to install Docker and Docker Compose: `curl https://get.docker.com | sh`

Add Your User to the Docker Group

Add your non-root user to the docker group so you can use Docker commands without sudo:

```
sudo usermod -aG docker <username>
```

Install Labdash in Docker

First we'll create a new folder to hold our applications and services files.

```
mkdir -p docker/labdash
```

Now we'll move into that folder.

```
cd docker/labdash
```

Here we'll create a new file in this location. It will be `compose.yaml`, which is our docker compose configuration file for the application.

```
---
services:
  lab-dash:
    container_name: lab-dash
    image: ghcr.io/anthonygress/lab-dash:latest
    privileged: true
    ports:
      - 2022:2022
    environment:
      - SECRET=<some random string of numbers and letters>
      # You can run `openssl rand -base64 32` to generate a key
    volumes:
      - ./sys:/sys:ro
      - ./config:/config
      - ./uploads:/app/public/uploads
      - /var/run/docker.sock:/var/run/docker.sock
    restart: unless-stopped
```

Notice any values in the file above surrounded by less than '<' and greater than '>' signs. These values you need to replace with actual values. In this particular docker compose file, you need to create a secret key. We can do this by exiting the nano editor with CTRL + x.

Next, run the command:

```
openssl rand -base64 32
```

Highlight and copy the value generated. Go back into your docker-compose.yml file with

```
nano docker-compose.yml
```

Move to the SECRET line, and clear any placeholder text. Now paste the value you just copied into the file. Save your changes with CTRL + O, then press Enter to confirm, and exit nano once again with CTRL + x.

Now we'll pull our images for Labdash with the command:

```
docker compose pull
```

Once it's pulled down, we can start our Labdash virtual machine with:

```
docker compose up -d.
```

I like to watch the logs the first time I start up a new docker virtual machine, to check for any errors I may need to correct. If you'd like to do that then use the command:

```
docker compose logs -f
```

after bringing up the virtual machine.

You can also just type them both in one line by concatenating the commands with `&&`.

```
docker compose up -d && docker compose logs -f
```

When you are done checking the logs, you can stop following them with CTRL + C.

Now, open your favorite browser, and go to the IP address of the **host** machine you are running the docker virtual machine on, and port 2022. This will bring you to the initial page of Labdash. You'll run through a short informational wizard, then it will prompt you to get signed up as the administrative user.

Once you're signed in, you can start building out a dashboard to help you quickly access all of your self-hosted services and applications, and to help you keep an eye on their status and all kinds of other great information available through a multitude of widgets.

Support my Channel and Content

Support my Channel and ongoing efforts through Patreon:

<https://patreon.com/awesomeopensource>

Paypal

<https://paypal.me/brianmcgonagill>

Merch

<https://osia.printify.me>