



Operating Manual

T-404-LOKA, Sleep DevOps, 2022-1
Reykjavik University, Menntavegi 1, IS-101 Reykjavik, Iceland

Ægir Máni Hauksson
aegir19@ru.is

May 13, 2022

Supervisors

Assistant Professor María Óskarsdóttir

Benedikt Hólm Þórðarson

Examiner

Assistant Professor Anna Sigríður Íslind

Dept. of Computer Science,
Reykjavík University

Contents

1	GitLab configuration	2
1.1	What are runners	2
1.2	Group Runner Interface	2
1.3	Project Runner Interface	2
1.4	Runner restrictions	3
1.5	Runner tags	3
1.6	Registration token	4
1.6.1	Group runner registration token	4
1.6.2	Project runner registration token	4
2	Server specification	4
2.1	Operating system	4
2.2	Requirements	5
2.3	Storage	5
2.4	RAM	5
2.5	CPU's	5
3	Runner	5
3.1	Configuration templates	5
3.2	Installation	6
3.2.1	Step 1: Connect to vpn.ru.is	6
3.2.2	Step 2: Login to machine	6
3.2.3	Step 3: Clone the SleepAgent project	6
3.2.4	Step 4: Navigate to directory	6
3.2.5	Step 5: Login to root user	6
3.2.6	Step 6: Run installation script	7
3.3	Modifying configuration	7
3.3.1	Finding the runner config	7
3.3.2	Modifying configs	7
3.3.3	Adding another runner	7

1 GitLab configuration

1.1 What are runners

GitLab Runner is an application that works with GitLab CI/CD to run jobs in a pipeline [4].

1.2 Group Runner Interface

Within a GitLab group [2], we can view all the available runners that projects within the group can use. To get to this view, navigate to CI/CD -> **Runners** in the left sidebar while inside the GitLab group. The state of each runner is presented. From this page, runners can be deleted, paused and edited by pressing the icons displayed on the right side of the listings.

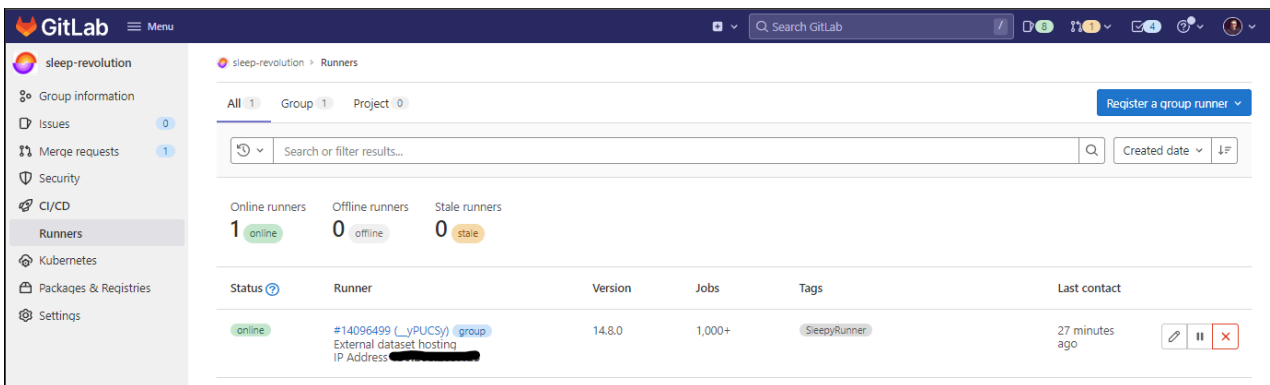


Figure 1: GitLab Group Runner Interface

By clicking one of the runners in the list, the runner details page can be viewed.

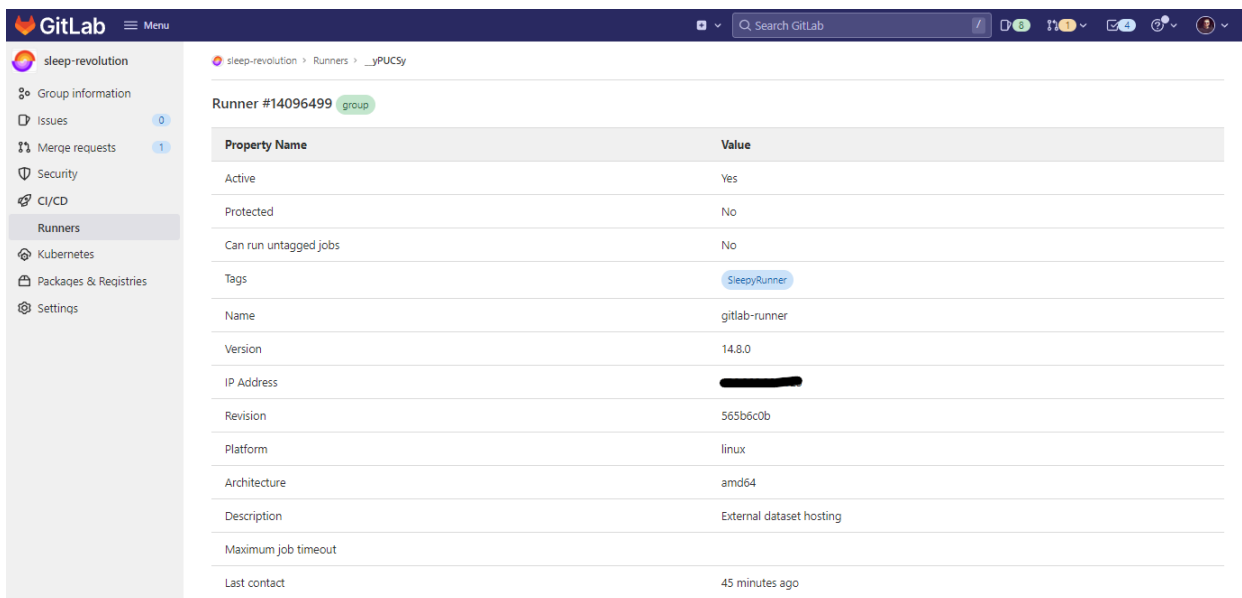


Figure 2: GitLab runner detail interface

1.3 Project Runner Interface

From within GitLab projects, the available runners can be viewed by navigating to CI/CD -> **Runners** in the left sidebar. The state of each runner is presented. If the runner was created by this project, it can be deleted, paused or edited from this view.

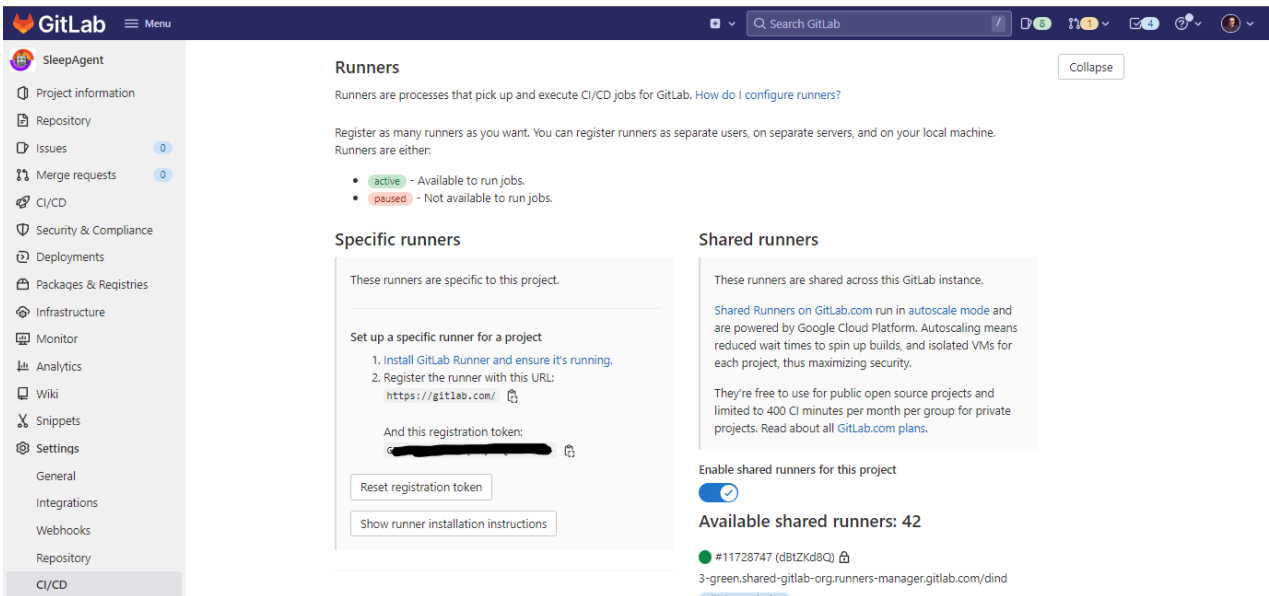


Figure 3: GitLab Project Runner Interface

1.4 Runner restrictions

Certain restrictions can be applied to runners through the runner editing page. In general, leaving the settings as in figure 4 will suffice.

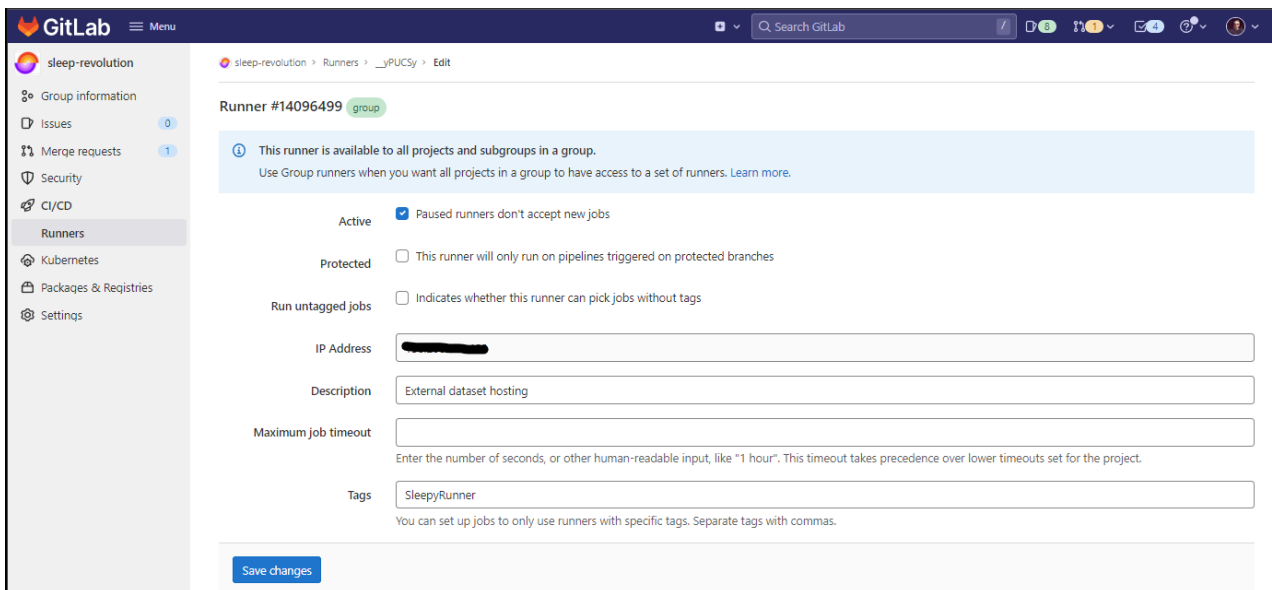


Figure 4: Runner restrictions

1.5 Runner tags

Runner tags are used to decide what runners are used in pipelines. The tags are referenced within pipeline configuration files.

Referencing runners in pipelines:

```
mytestjob:
  stage: test
  tags: ["SleepyRunner", "WakyRunner"]
```

1.6 Registration token

1.6.1 Group runner registration token

Registering a runner through the GitLab Group [2] will make a runner available to all projects. The token which is required to register a runner with the group is retrievable from within the group runner interface (1.2). The image below highlights the tab used to copy the token.

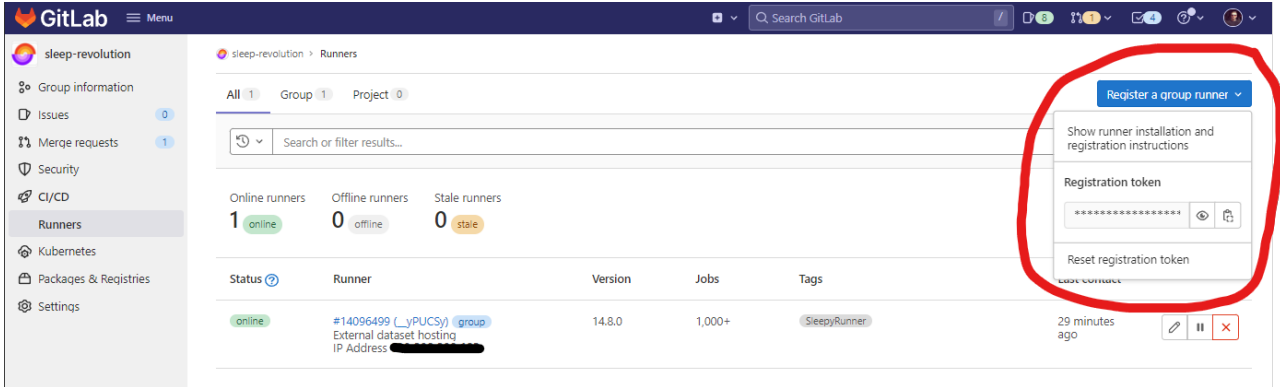


Figure 5: Group Runner Registration Token

1.6.2 Project runner registration token

Registering a runner through a GitLab project will make a runner available to the project only. The token which is required to register a runner with the project is retrievable from within the project runner interface (1.3). The image below highlights the tab used to copy the token.

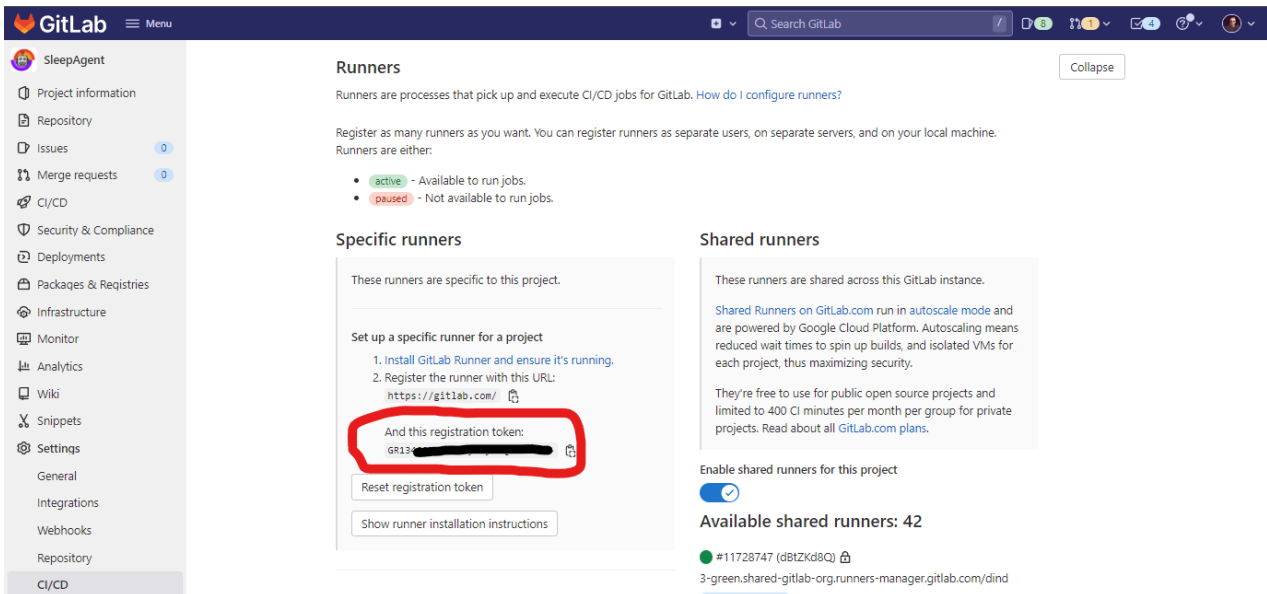


Figure 6: Project Runner Registration Token

2 Server specification

2.1 Operating system

The server must run on a Linux or preferably Ubuntu Server [8] operating system. This is because the gitlab-runner [4] software binaries are compiled for Linux-x86-64.

2.2 Requirements

The following are the requirements that must be met in order to install and manage GitLab Runner on the server.

1. Root User Access Privileges
why: To manage and install GitLab Runner
how: Contact cluster administration
2. Git
why: To clone the SleepAgent repository
how: `$ sudo apt-get install git`
3. cURL
why: To download software binaries in the installation scripts
how: `$ sudo apt-get install curl`

2.3 Storage

The amount of storage required depends on the use case of the runner. Generally, having access to 50GB of storage will be enough to ensure future needs. This storage requirement comes mainly from the need to store docker images and temporary artifacts from pipelines. If a large variety of docker images are being used on the pipelines ran by the runner, configuring additional storage is recommended.

2.4 RAM

The minimum amount of RAM recommended for the system is 4GB. If the runner is to be used to process large data sets in pipelines, 16GB of RAM or more should be used. Communicating with the intended users of the runner should help with deciding on how much RAM needs to be configured. Sharing the RAM with other virtual machines should be fine, but in that case it should be made sure that the RAM needs of the pipelines can be met within a reasonable amount of time.

2.5 CPU's

Having more CPU's will mean the pipelines are executed faster. Having 4 CPU's will be enough for most use cases, and can be configured to be shared with other virtual machines. Sharing the CPU's will prevent holding unused resources, since pipelines are only being used a small portion of the time.

3 Runner

3.1 Configuration templates

GitLab runner is configured through `config.toml` files. Some GitLab runner config files have been created as templates within the SleepAgent GitLab repository at `scripts/templates`. They are used by the installation script to configure the initial `config.toml` file for the runner. The following are their names and usages.

1. `default.toml`: Default docker runner configuration
2. `sleepy_docker.toml`: Arora docker runner configuration

3.2 Installation

3.2.1 Step 1: Connect to vpn.ru.is

Using a VPN client such as Cisco Anyconnect [1] or OpenConnect [7], establish a secure connection to `vpn.ru.is` and use your Sleep Revolution credentials for authentication.

3.2.2 Step 2: Login to machine

Login to the Reykjavík University server machine that you wish to install the runner on. If you already have a runner set up on the machine, refer to section 3.3.3 for instructions on how to add additional runners.

3.2.3 Step 3: Clone the SleepAgent project

To execute our installation script we need to clone the SleepAgent [9] project. If you do not have access to this project please request it from the Sleep Revolution administrators at Reykjavík University.

In order to authenticate when cloning the project, you need to link a public SSH key on your server to your GitLab account [6]. If you do not wish to have an SSH key linked to your GitLab account on the server for security reasons or otherwise, you can create a Personal Access Token [3] with `read_repository` permissions. Using a personal access token allows you to clone the project with one-time authentication.

Clone with a personal access token:

```
$ git clone https://<name>:<value>/sleep-revolution/sleepagent.git
```

Clone with ssh key configured:

```
$ git clone git@gitlab.com:sleep-revolution/sleepagent.git
```

3.2.4 Step 4: Navigate to directory

The installation script resides within the SleepAgent project in the `scripts/` folder. Navigate to the `scripts/` folder in terminal.

Terminal command:

```
$ cd sleepagent/scripts
```

3.2.5 Step 5: Login to root user

GitLab Runner can only be started with root user privileges. The installation script will trigger this action before terminating.

Login command:

```
$ sudo su
> Enter the root user password
```

3.2.6 Step 6: Run installation script

The script takes care of the installation and registration of your runner. The script will prompt for the following inputs in the given order.

1. Runner Registration Token: See section 1.6
2. Runner Tags: See section 1.5
3. Runner Name: Any name or description you wish to use for the runner
4. Configuration Template: See section 3.1

Installation command:

```
$ bash setup_runner.sh docker
> Enter Runner Registration Token
> Enter Runner Tags (comma separated if many)
> Enter Runner Name
> Enter Template filename (with .toml)
```

3.3 Modifying configuration

3.3.1 Finding the runner config

The GitLab Runner configuration is managed through a file called `config.toml`. This file is located on the server where the runner is installed at, in the directory `/etc/gitlab-runner`.

3.3.2 Modifying configs

To learn how to configure the `config.toml` file please read up on the following reference [5] provided by GitLab on advanced runner configuration. Topics of special interest might include resource limitations and cache storage configuration.

3.3.3 Adding another runner

If you already have a runner configured on your server and want to add another one, registering a new one is enough. To do this, we use a `register.sh` script residing within the SleepAgent GitLab repository. To run the registration, first navigate to where you cloned the SleepAgent repository. The script requires the following arguments in the given order.

1. Runner Registration Token: See section 1.6
2. Runner Tags: See section 1.5
3. Runner Name: Any name you wish to use for the runner

Registration command:

```
$ bash scripts/docker/register.sh <token> <tags> <name>
```

Once the new runner is registered, GitLab Runner adds a default configuration inside the `config.toml` file (3.3.1) for the new runner. You will want to edit this in a similar manner to our template configurations (3.1).

References

- [1] Cisco. Cisco anyconnect - vpn software. <https://www.cisco.com/c/en/us/support/security/anyconnect-secure-mobility-client-v4-x/model.html>. Accessed: 2022-5-10.
- [2] GitLab. Gitlab - gitlab groups. <https://docs.gitlab.com/ee/user/group/>. Accessed: 2022-5-12.
- [3] GitLab. Gitlab - personal access token. https://docs.gitlab.com/ee/user/profile/personal_access_tokens.html. Accessed: 2022-5-11.
- [4] GitLab. Gitlab - runner. <https://docs.gitlab.com/runner/>. Accessed: 2022-5-11.
- [5] GitLab. Gitlab - runner advanced configuration. <https://docs.gitlab.com/runner/configuration/advanced-configuration.html>. Accessed: 2022-5-11.
- [6] GitLab. Gitlab - ssh cloning. <https://docs.gitlab.com/ee/user/ssh.html>. Accessed: 2022-5-11.
- [7] OpenConnect. Openconnect - vpn software. <http://www.infradead.org/openconnect/>. Accessed: 2022-5-10.
- [8] Ubuntu. Ubuntu - server iso download. <https://ubuntu.com/download/server>. Accessed: 2022-5-11.
- [9] Ægir Máni Hauksson. Sleepagent gitlab project. <https://gitlab.com/sleep-revolution/sleepagent>. Accessed: 2022-5-11.