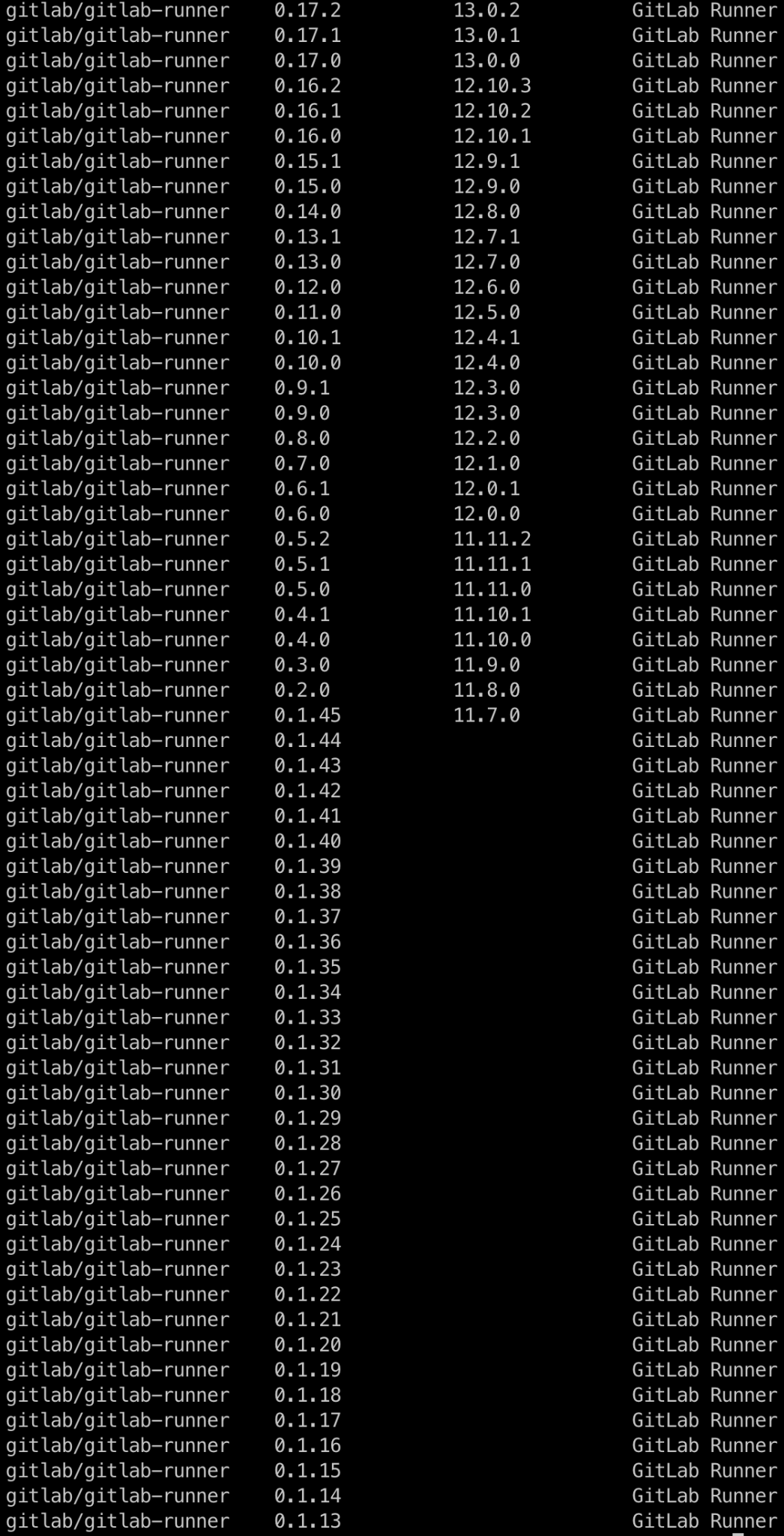
Развернуть gitlab раннеров в kubernetes кластере и настроить ci/cd

Чтобы развернуть раннеров использовал helm и helmfile.

Текущая версия GitLab Community Edition 11.0.0-rc12

Нашел чарт gitlab раннера

helm search repo -l gitlab/gitlab-runner



Нужной версии гитлаба не было в списке

helm show values gitlab/gitlab-runner --version 0.1.15 | yq '.image'

Так и не смог найти раннер для версии GitLab Community Edition 11.0.0-rc12, но решил взять 0.1.34

Поставил раннера в кластер

helm install --namespace runner gitlab-runner -f values.yaml gitlab/gitlab-runner --version 0.1.34

Получил ошибку

Error: INSTALLATION FAILED: unable to build kubernetes objects from release manifest: resource mapping not found for name: "gitlab-runner-gitlab-runner" namespace: "" from "": no matches for kind "Deployment" in version "extensions/v1beta1"

ensure CRDs are installed first

Она связана со старым чартом, который не поддерживается новым кубернетисом

Выкачал чарт и изменил

apiVersion: extensions/v1beta1

на

apiVersion: apps/v1

Сделал helmfile(1) для раннера в репе с чартом и настроил values.yaml(2) и secrets.yaml(3) для 3х раннеров: дефолтный, привилегированный и раннер с доступом к кубу

Далее

helmfile apply

Далее нужно настроить ci так, чтобы при push срабатывал плагин helmfile diff, при merge срабатывал плагин helmfile apply (кластер должен соответствовать репе).

Создал файл .gitlab-ci.yml

Хотел использовать runners.kubernetes.pod\_spec в config.toml чтобы под, создаваемый раннером имел доступ к кубу

<https://docs.gitlab.com/runner/executors/kubernetes.html>

Overwrite generated pod specifications (Beta)

Introduced in GitLab Runner 15.10.

Так как у наc 11.3.0 пришлось искать другой способ передать роль создаваемому раннеру

С помощью роли сделал конфиг и передал его через переменные в гитлабе

Настроил для ci создание файла config в папке ~/.kube из переданной переменной

- mkdir $HOME/.kube

- echo $KUBECONFIG\_CONTENT > $HOME/.kube/config

Гитлаб переформатировал файл в строчку, из-за чего он стал нечитаемым. Чтобы передать файл строчкой закодировал его base64 и передал получившуюся строчку, а уже потом в джобе прописал декодирование этой строчки

Далее передал ключ sops, которым зашифрованы секреты.

У раннера есть доступ в куб (работает kubectl) и возможность использовать helmfile