

Создание пайплайна в GitlabCI

--

Для начала работ вам понадобится:

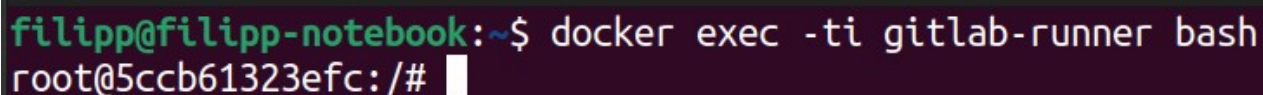
- *docker*
- *gitlab*
- *gitlab-runner*

Подробная установка этих инструментов рассмотрена в предыдущих уроках

--

Выполним изменение используемого образа в запущенном `gitlab-runner`:

```
$ docker exec -ti gitlab-runner bash
```

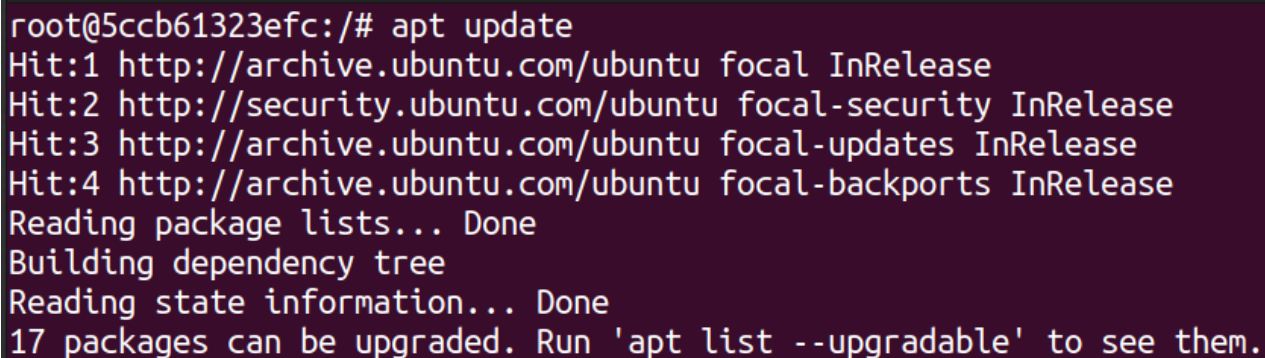


```
filipp@filipp-notebook:~$ docker exec -ti gitlab-runner bash
root@5ccb61323efc:/#
```

Находясь в псевдо-терминале контейнера, изменим конфигурационный файл `gitlab-runner`.

Если в контейнере `gitlab-runner` отсутствует текстовый редактор, установите его:

```
$ apt update
```



```
root@5ccb61323efc:/# apt update
Hit:1 http://archive.ubuntu.com/ubuntu focal InRelease
Hit:2 http://security.ubuntu.com/ubuntu focal-security InRelease
Hit:3 http://archive.ubuntu.com/ubuntu focal-updates InRelease
Hit:4 http://archive.ubuntu.com/ubuntu focal-backports InRelease
Reading package lists... Done
Building dependency tree
Reading state information... Done
17 packages can be upgraded. Run 'apt list --upgradable' to see them.
```

```
$ apt install nano
```

```
root@5ccb61323efc:/# apt install nano
Reading package lists... Done
Building dependency tree
Reading state information... Done
Suggested packages:
  hunspell
The following NEW packages will be installed:
  nano
0 upgraded, 1 newly installed, 0 to remove and 17 not upgraded.
Need to get 269 kB of archives.
After this operation, 868 kB of additional disk space will be used.
Get:1 http://archive.ubuntu.com/ubuntu focal/main amd64 nano amd64 4.8-
Fetched 269 kB in 1s (190 kB/s)
debconf: delaying package configuration, since apt-utils is not installed
Selecting previously unselected package nano.
(Reading database ... 9547 files and directories currently installed.)
Preparing to unpack .../nano_4.8-1ubuntu1_amd64.deb ...
Unpacking nano (4.8-1ubuntu1) ...
Setting up nano (4.8-1ubuntu1) ...
update-alternatives: using /bin/nano to provide /usr/bin/editor (editor)
update-alternatives: warning: skip creation of /usr/share/man/man1/editor.1.gz (of link group editor) doesn't exist
update-alternatives: using /bin/nano to provide /usr/bin/pico (pico) in
update-alternatives: warning: skip creation of /usr/share/man/man1/pico.1.gz (of link group pico) doesn't exist
```

Присутствуем к модификации конфигурационного файла:

```
$ nano /etc/gitlab-runner/config.toml
```

```
root@5ccb61323efc:/# nano /etc/gitlab-runner/config.toml
```

Изменяем образ "alpine:3.17" на образ "docker:20.10.16"

В поле "volumes" добавляем к монтированию кэша еще запись о монтировании сокета докера (внимание! эта конфигурация не годится для production-ready использования):

```
volumes = ["/var/run/docker.sock:/var/run/docker.sock", "/cache"]
```

```

name = "5ccb61323efc"
url = "http://172.17.0.2:8929/"
id = 1
token = "GMU589TxDJMv_snsUKZb"
token_obtained_at = 2023-02-17T11:14:30Z
token_expires_at = 0001-01-01T00:00:00Z
executor = "docker"
[runners.custom_build_dir]
[runners.cache]
  MaxUploadedArchiveSize = 0
[runners.cache.s3]
[runners.cache.gcs]
[runners.cache.azure]
[runners.docker]
  tls_verify = false
  image = "docker:20.10.16"
  privileged = false
  disable_entrypoint_overwrite = false
  oom_kill_disable = false
  disable_cache = false
  volumes = ["/var/run/docker.sock:/var/run/docker.sock", ["/cache"]]
  shm_size = 0

```

Сохраняем файл конфигурации и выходим из псевдо-терминала контейнера gitlab-runner

--

Создадим новый проект в gitlab под названием "hello-docker":



Create blank project

Create a blank project to store your files, plan your work, and collaborate on code, among other things.

New project > Create blank project

Project name

Project URL

Project slug

Want to organize several dependent projects under the same namespace? [Create a group](#).

Visibility Level

☒ Private

Project access must be granted explicitly to each user. If this project is part of a group, access is granted to members of the group.

☐ Internal

The project can be accessed by any logged in user except external users.

☐ Public

The project can be accessed without any authentication.

Project Configuration

☒ Initialize repository with a README

Allows you to immediately clone this project's repository. Skip this if you plan to push up an existing repository.

☐ Enable Static Application Security Testing (SAST)

Analyze your source code for known security vulnerabilities. [Learn more](#).

Create project

Cancel

Сохраним URL проекта (например, из адресной строки браузера)

Задействуем приложение, на примере которого мы в предыдущих уроках демонстрировали возможности многоэтапной сборки, немного его модифицировав.

В терминале, находясь в директории проекта, выполним следующие команды:

Инициализация репозитория:

```
$ git init --initial-branch=main
```

```
filipp@filipp-notebook:~/Desktop/hello-docker$ git init --initial-branch=main  
Initialized empty Git repository in /home/filipp/Desktop/hello-docker/.git/
```

Далее выполним добавление файла с описанием пайплайна (.gitlab-ci.yml):

```
$ nano .gitlab-ci.yml
```

И наполним его следующим содержанием:

```
stages:  
  - build  
  - network-create  
  - run  
  
build-job:  
  stage: build  
  script:  
    - docker build . -t hello-world-multistage  
  
network-create-job:  
  stage: network-create  
  script:  
    - docker network create --subnet=172.19.0.0/16 hw-network  
  
run-job:  
  stage: run  
  script:  
    - docker run --net hw-network --ip 172.19.0.9 --name hello-  
world -d hello-world-multistage
```

```

GNU nano 6.2                               .gitlab-ci.yml *
stages:
- build
- network-create
- run

build-job:
  stage: build
  script:
    - docker build . -t hello-world-multistage

network-create-job:
  stage: network-create
  script:
    - docker network create --subnet=172.19.0.0/16 hw-network

run-job:
  stage: run
  script:
    - docker run --net hw-network --ip 172.19.0.9 --name hello-world -d h>

```

Сохраним файл и привяжем локальный репозиторий с удаленным (используйте ранее скопированный URL проекта в gitlab):

```
$ git remote add origin http://172.17.0.2:8929/gitlab-instance-6435c815/hello-docker.git
```

```

filipp@filipp-notebook:~/Desktop/hello-docker$ git remote add origin http://172.17.0.2:8929/gitlab-instance-6435c815/hello-docker.git

```

Сохраняем изменения проекта, индексируем, делаем коммит и отправляем в удаленный репозиторий:

```

$ git add .
$ git commit -m 'initial commit'
$ git push -u origin --all


```

```





filipp@filipp-notebook:~/Desktop/hello-docker$ git add .
filipp@filipp-notebook:~/Desktop/hello-docker$ git commit -m 'initial commit'
[main (root-commit) a3869b0] initial commit
5 files changed, 66 insertions(+)
create mode 100644 .gitlab-ci.yml
create mode 100755 Dockerfile
create mode 100755 go.mod
create mode 100755 go.sum
create mode 100755 main.go
filipp@filipp-notebook:~/Desktop/hello-docker$ git push -u origin --all
Username for 'http://172.17.0.2:8929': root
Password for 'http://root@172.17.0.2:8929':
Enumerating objects: 7, done.
Counting objects: 100% (7/7), done.
Delta compression using up to 8 threads
Compressing objects: 100% (7/7), done.
Writing objects: 100% (7/7), 1.52 KiB | 1.52 MiB/s, done.
Total 7 (delta 0), reused 0 (delta 0), pack-reused 0
To http://172.17.0.2:8929/gitlab-instance-6435c815/hello-docker.git
 * [new branch]      main -> main
Branch 'main' set up to track remote branch 'main' from 'origin'.

```


Проверим, что содержимое проекта успешно сохранено в удаленном репозитории и пайплайн запущен:



hello-docker

Project ID: 3



 Star 0
 Fork 0

1 Commit
1 Branch
0 Tags
102 KB Project Storage







Initial commit
filipp authored 2 minutes ago


a3869b0a

main
hello-docker /
Find file
Web IDE
Clone

CI/CD configuration
Add README
Add LICENSE
Add CHANGELOG
Add CONTRIBUTING

Auto DevOps enabled
Add Kubernetes cluster
Configure Integrations

Name	Last commit	Last update
 .gitlab-ci.yml	initial commit	2 minutes ago
 Dockerfile	initial commit	2 minutes ago
 go.mod	initial commit	2 minutes ago
 go.sum	initial commit	2 minutes ago
 main.go	initial commit	2 minutes ago

Изменения в проект успешно внесены

GitLab Instance > hello-docker > Pipelines

All 2	Finished	Branches	Tags	Clear runner caches	CI lint	Run pipeline
Filter pipelines				Q	Show Pipeline ID v	
Status	Pipeline	Triggerer	Stages			
<div>passed</div> <div>00:00:21 just now</div>	Initial commit #11 main a3869b0a		<div> <div></div> <div></div> <div></div> </div>	<div> <div></div> <div></div> <div></div> </div>		

Пайплайн запущен, выполнен и успешно завершен

Перейдем в терминал и выполним команду првоерки работоспособности развернутого приложения:

```
$ curl http://172.19.0.9:8080
```

```
filipp@filipp-notebook:~/Desktop/hello-docker$ curl http://172.19.0.9:8080
Hello there!
```

Приложение успешно развернуто и работает в штатном режиме

--