

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ
РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«САМАРСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
УНИВЕРСИТЕТ ИМЕНИ АКАДЕМИКА С.П. КОРОЛЕВА»

Институт информатики и кибернетики
Кафедра геоинформатики и информационной безопасности

Отчёт по лабораторной работе №2
«Git and CI»

Выполнила: Беседина В.М.

Группа: 6412-100503D

Проверил: Авдеев Е.В.

Camapa 2023

Steps

1. Gitlab setup (on VM/Docker/Minikube) (VM variant) Download Gitlab-Bitnami vm image from <https://bitnami.com/stack/gitlab/virtual-machine>

2. Create repo in gitlab with sources of your app

Upload <https://github.com/olindata/sample-gitlabci-cpp-project> to your Gitlab server.

3. Setup runner (Gitlab, Docker, Jenkins, ArgoCD)

4. Edit `.gitlab-ci.yml` to run runner in shell mode (without Docker)

5. Run Pipeline: CI/CD > Pipelines > Run pipeline

6. Create Assignment2 report and send it by e-mail (docx/link to google doc) or through creation repo fork + pull request.

Steps 1.

```
bitnami-gitlab-ce-15-15.9.3-ce.0-r0-debian-11-amd64 [Работает] - Oracle VM VirtualBox
Файл  Машина  Вид  Ввод  Устройства  Справка

*** Welcome to GitLab CE packaged by Bitnami ***
*** Built using Debian GNU/Linux 11 (bullseye) - Kernel 5.10.0-21-amd64 (tty1) ***

*** You can access the application at http://192.168.0.168 ***
*** The default username is 'root' and the default password is 'uA7IzTwyqfLq' ***
*** You can find out more at https://docs.bitnami.com/virtual-machine/apps/gitlab/ ***
*** If you find any issues, please visit https://github.com/bitnami/vms/issues ***

*****
To access the console, login with user 'bitnami' and password 'bitnami'
*****

debian login: bitnami
Password:
Linux debian 5.10.0-21-amd64 #1 SMP Debian 5.10.162-1 (2023-01-21) x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.

  _ _ _ _ _
 | _ | _ | _ | _ | _ | _ |
 | _ | _ | _ | _ | _ | _ |
 | _ | _ | _ | _ | _ | _ |

-> Welcome to GitLab CE packaged by Bitnami 15.9.3-ce.0
-> Documentation:  https://docs.bitnami.com/virtual-machine/apps/gitlab/
-> Bitnami Support: https://github.com/bitnami/vms/issues

For security reasons, the password will be changed
Changing password for bitnami.
Current password:
passwd: Authentication token manipulation error
passwd: password unchanged
bitnami@debian:~$

bitnami@debian:~$ sudo rm -f /etc/ssh/sshd_not_to_be_run
bitnami@debian:~$ sudo systemctl enable ssh
Synchronizing state of ssh.service with SysV service script with /lib/systemd/systemd-sysv-install.
Executing: /lib/systemd/systemd-sysv-install enable ssh
bitnami@debian:~$ sudo systemctl start ssh
bitnami@debian:~$ sudo systemctl status ssh
■ ssh.service - OpenBSD Secure Shell server
   Loaded: loaded (/lib/systemd/system/ssh.service; enabled; vendor preset: enabled)
   Active: active (running) since Sun 2023-03-12 18:55:52 UTC; 27s ago
     Docs: man:sshd(8)
           man:sshd_config(5)
   Process: 2217 ExecStartPre=/usr/sbin/sshd -t (code=exited, status=0/SUCCESS)
   Main PID: 2218 (sshd)
     Tasks: 1 (limit: 4678)
    Memory: 2.0M
       CPU: 15ms
   CGroup: /system.slice/ssh.service
           └─2218 sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups

Mar 12 18:55:52 debian systemd[1]: Starting OpenBSD Secure Shell server...
Mar 12 18:55:52 debian sshd[2218]: Server listening on 0.0.0.0 port 22.
Mar 12 18:55:52 debian sshd[2218]: Server listening on :: port 22.
Mar 12 18:55:52 debian systemd[1]: Started OpenBSD Secure Shell server.
bitnami@debian:~$

bitnami@debian:~$ sudo /etc/init.d/ssh force-reload
Reloading ssh configuration (via systemctl): ssh.service.
bitnami@debian:~$
```

```
bitnami-gitlab-ce-15-15.9.3-ce.0-r0-debian-11-amd64 [Работает] - Oracle VM VirtualBox
Файл  Машина  Вид  Ввод  Устройства  Справка
GNU nano 5.4 /etc/ssh/sshd_config *
# Expect .ssh/authorized_keys2 to be disregarded by default in future.
#AuthorizedKeysFile .ssh/authorized_keys .ssh/authorized_keys2

#AuthorizedPrincipalsFile none

#AuthorizedKeysCommand none
#AuthorizedKeysCommandUser nobody

# For this to work you will also need host keys in /etc/ssh/ssh_known_hosts
#HostbasedAuthentication no
# Change to yes if you don't trust ~/.ssh/known_hosts for
# HostbasedAuthentication
#IgnoreUserKnownHosts no
# Don't read the user's ~/.rhosts and ~/.shosts files
#IgnoreRhosts yes

# To disable tunneled clear text passwords, change to no here!
PasswordAuthentication yes_
#PermitEmptyPasswords no

# Change to yes to enable challenge-response passwords (beware issues with
# some PAM modules and threads)
ChallengeResponseAuthentication no

# Kerberos options
#KerberosAuthentication no
#KerberosOrLocalPasswd yes
#KerberosTicketCleanup yes
#KerberosGetAFSToken no

# GSSAPI options
#GSSAPIAuthentication no
#GSSAPICleanupCredentials yes

[ Search Wrapped ]
^G Help      ^O Write Out  ^W Where Is   ^K Cut        ^T Execute    ^C Location   ^U Undo
^X Exit      ^R Read File  ^N Replace    ^U Paste      ^J Justify    ^_ Go To Line   ^E Redo
```

```
bitnami@debian:~$ wget https://github.com/olindata/sample-gitlabci-cpp-project/archive/refs/heads/master.zip
--2023-03-12 20:38:19-- https://github.com/olindata/sample-gitlabci-cpp-project/archive/refs/heads/master.zip
Resolving github.com (github.com)... 140.82.121.4
Connecting to github.com (github.com)|140.82.121.4|:443... connected.
HTTP request sent, awaiting response... 302 Found
Location: https://codeload.github.com/olindata/sample-gitlabci-cpp-project/zip/refs/heads/master [following]
--2023-03-12 20:38:20-- https://codeload.github.com/olindata/sample-gitlabci-cpp-project/zip/refs/heads/master
Resolving codeload.github.com (codeload.github.com)... 140.82.121.10
Connecting to codeload.github.com (codeload.github.com)|140.82.121.10|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: unspecified [application/zip]
Saving to: 'master.zip'

master.zip          [ <=> ] 2.05K --.-KB/s  in 0s

2023-03-12 20:38:20 (23.7 MB/s) - 'master.zip' saved [2095]

bitnami@debian:~$ unzip master.zip
Archive: master.zip
4af9cf42b539a1a094014ff326b3b3bad10f189a
  creating: sample-gitlabci-cpp-project-master/
  extracting: sample-gitlabci-cpp-project-master/.gitignore
  inflating: sample-gitlabci-cpp-project-master/.gitlab-ci.yml
  inflating: sample-gitlabci-cpp-project-master/README.md
  inflating: sample-gitlabci-cpp-project-master/helloworld.cpp
  inflating: sample-gitlabci-cpp-project-master/verify.sh
bitnami@debian:~$
```

Step 2

```
bitnami@debian:~/sample-gitlabci-cpp-project-master$ git commit -m "initial commit"
[master (root-commit) 80639e1] initial commit
5 files changed, 66 insertions(+)
create mode 100644 .gitignore
create mode 100644 .gitlab-ci.yml
create mode 100644 README.md
create mode 100644 helloworld.cpp
create mode 100755 verify.sh
bitnami@debian:~/sample-gitlabci-cpp-project-master$ git push -u origin --all
Username for 'https://192.168.0.168': root
Password for 'https://root@192.168.0.168':
Enumerating objects: 7, done.
Counting objects: 100% (7/7), done.
Compressing objects: 100% (6/6), done.
Writing objects: 100% (7/7), 1.28 KiB | 1.28 MiB/s, done.
Total 7 (delta 0), reused 0 (delta 0), pack-reused 0
remote:
remote:
remote: The private project root/sample-gitlabci-cpp-project was successfully created.
remote:
remote: To configure the remote, run:
remote:   git remote add origin https://192.168.0.168/root/sample-gitlabci-cpp-project.git
remote:
remote: To view the project, visit:
remote:   https://192.168.0.168/root/sample-gitlabci-cpp-project
remote:
remote:
remote: To https://192.168.0.168/root/sample-gitlabci-cpp-project.git
* [new branch]      master -> master
Branch 'master' set up to track remote branch 'master' from 'origin'.
bitnami@debian:~/sample-gitlabci-cpp-project-master$ _
```

Administrator > sample-gitlabci-cpp-project

S

sample-gitlabci-cpp-project
Project ID: 2

Star 0 Fork 0

1 Commit

1 Branch

0 Tags

102 KB Project Storage

initial commit
lera authored in 3 hours

80639e11

master

sample-gitlabci-cpp-project /

Find file Web IDE Clone

README CI/CD configuration Add LICENSE Add CHANGELOG Add CONTRIBUTING

Auto DevOps enabled

Add Kubernetes cluster Add Wiki Configure Integrations

Name	Last commit	Last update
.gitignore	initial commit	in 3 hours
.gitlab-ci.yml	initial commit	in 3 hours
README.md	initial commit	in 3 hours
helloworld.cpp	initial commit	in 3 hours
verify.sh	initial commit	in 3 hours

README.md

Gitlab Sample C++ Project

Documentation

Simply clone this repository in your Gitlab instance, and make sure to have a Gitlab runner defined (preferably with a docker-based executor) and you'll see your gitlab ci builds working!

Need Help?

Need help? Feel free to contact Farley at OlinData dot com

or

Step 3

```
bitnami@debian:~/sample-gitlabci-cpp-project-master$ curl -L "https://packages.gitlab.com/install/repositories/runner/gitlab-runner/script.deb.sh" | sudo bash
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left   Speed
100 6885 100 6885    0     0 17790      0 --:--:-- --:--:-- --:--:-- 17790
Detected operating system as debian/bullseye.
Checking for curl...
Detected curl...
Checking for gpg...
Detected gpg...
Running apt-get update... done.
Installing debian-archive-keyring which is needed for installing
apt-transport-https on many Debian systems.
Installing apt-transport-https... done.
Installing /etc/apt/sources.list.d/runner_gitlab-runner.list...done.
Importing packagecloud gpg key... done.
Running apt-get update... done.
```

```
bitnami@debian:~/sample-gitlabci-cpp-project-master$ sudo apt install gitlab-runner
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Suggested packages:
  docker-engine
The following NEW packages will be installed:
  gitlab-runner
0 upgraded, 1 newly installed, 0 to remove and 0 not upgraded.
Need to get 449 MB of archives.
After this operation, 488 MB of additional disk space will be used.
Get:1 https://packages.gitlab.com/runner/gitlab-runner/debian bullseye/main amd64 gitlab-runner amd64 15.9.1 [449 MB]
Fetched 449 MB in 22s (20.4 MB/s)
debconf: delaying package configuration, since apt-utils is not installed
Selecting previously unselected package gitlab-runner.
(Reading database ... 110811 files and directories currently installed.)
Preparing to unpack .../gitlab-runner_15.9.1_amd64.deb ...
Unpacking gitlab-runner (15.9.1) ...
Setting up gitlab-runner (15.9.1) ...
GitLab Runner: creating gitlab-runner...
Home directory skeleton not used
Runtime platform                                arch=amd64 os=linux pid=9623 revision=d540b510 version=15.9.1
gitlab-runner: the service is not installed
Runtime platform                                arch=amd64 os=linux pid=9629 revision=d540b510 version=15.9.1
gitlab-ci-multi-runner: the service is not installed
Runtime platform                                arch=amd64 os=linux pid=9645 revision=d540b510 version=15.9.1
Runtime platform                                arch=amd64 os=linux pid=9684 revision=d540b510 version=15.9.1
INFO: Docker installation not found, skipping clear-docker-cache
bitnami@debian:~/sample-gitlabci-cpp-project-master$
```

```
Prerequisites
bitnami-gitlab-ce-15-15.9.3-ce.0-r0-debian-11-amd64 [Работаer] - Oracle VM VirtualBox
Файл Машина Вид Ввод Устройства Справка
GNU nano 5.4 /etc/gitlab/gitlab.rb *
# web_server['username'] = 'gitlab-www'
# web_server['group'] = 'gitlab-www'
# web_server['uid'] = nil
# web_server['gid'] = nil
# web_server['shell'] = '/bin/false'
# web_server['home'] = '/var/opt/gitlab/nginx'

#####
## GitLab NGINX
##! Docs: https://docs.gitlab.com/omnibus/settings/nginx.html
#####

# nginx['enable'] = true
# nginx['client_max_body_size'] = '250m'
# nginx['redirect_http_to_https'] = false
# nginx['redirect_http_to_https_port'] = 80

##! Most root CA's are included by default
# nginx['ssl_client_certificate'] = "/etc/gitlab/ssl/ca.crt"

##! enable/disable 2-way SSL client authentication
# nginx['ssl_verify_client'] = "off"

##! if ssl_verify_client on, verification depth in the client certificates chain
# nginx['ssl_verify_depth'] = "1"

# nginx['ssl_certificate'] = '/etc/gitlab/ssl/server.crt'
# nginx['ssl_certificate_key'] = '/etc/gitlab/ssl/server.key'
# nginx['ssl_ciphers'] = "ECDHE-ECDSA-AES128-GCM-SHA256:ECDHE-RSA-AES128-GCM-SHA256:ECDHE-ECDSA-AES"
# nginx['ssl_prefer_server_ciphers'] = "off"

##! **Recommended by: https://raymii.org/s/tutorials/Strong_SSL_Security_On_nginx.html
##! https://cipherli.st/**

^G Help      ^O Write Out ^W Where Is  ^K Cut       ^T Execute   ^C Location  ^U Undo
^X Exit      ^R Read File ^M Replace   ^U Paste     ^J Justify   ^G Go To Line ^E Redo

Right Ctrl
```

```
bitnami-gitlab-ce-15-15.9.3-ce.0-r0-debian-11-amd64 [Работаer] - Oracle VM VirtualBox
Файл Машина Вид Ввод Устройства Справка
* file[/var/opt/gitlab/crond/database-reindexing] action delete (up to date)
(up to date)
[2023-03-13T01:07:17+00:00] INFO: templatesymlink[Create a gitlab.yml and create a symlink to Rails
root] sending restart action to runit_service[puma] (delayed)
Recipe: gitlab::puma
* runit_service[puma] action restart (up to date)
[2023-03-13T01:07:17+00:00] INFO: templatesymlink[Create a gitlab.yml and create a symlink to Rails
root] sending restart action to sidekiq_service[sidekiq] (delayed)
Recipe: gitlab::sidekiq
* sidekiq_service[sidekiq] action restart
* service[sidekiq] action nothing (skipped due to action :nothing)
* runit_service[sidekiq] action restart (up to date)
(up to date)
[2023-03-13T01:07:22+00:00] INFO: templatesymlink[Create a gitlab.yml and create a symlink to Rails
root] sending run action to execute[clear the gitlab-rails cache] (delayed)
Recipe: gitlab::gitlab-rails
* execute[clear the gitlab-rails cache] action run[2023-03-13T01:08:31+00:00] INFO: execute[clear
the gitlab-rails cache] ran successfully

- execute /opt/gitlab/bin/gitlab-rake cache:clear
[2023-03-13T01:08:31+00:00] INFO: template[/var/opt/gitlab/gitlab-kas/gitlab-kas-config.yml] sending
restart action to runit_service[gitlab-kas] (delayed)
Recipe: gitlab-kas::enable
* runit_service[gitlab-kas] action restart (up to date)
[2023-03-13T01:08:41+00:00] INFO: template[/var/opt/gitlab/nginx/conf/gitlab-http.conf] sending rest
art action to runit_service[nginx] (delayed)
Recipe: nginx::enable
* runit_service[nginx] action restart (up to date)
[2023-03-13T01:08:41+00:00] INFO: Cinc Client Run complete in 98.929926527 seconds

Running handlers:
[2023-03-13T01:08:41+00:00] INFO: Running report handlers
Running handlers complete
[2023-03-13T01:08:42+00:00] INFO: Report handlers complete
Infra Phase complete, 6/795 resources updated in 01 minutes 40 seconds
gitlab Reconfigured!
bitnami@debian:~/sample-gitlabci-cpp-project-master$ _
```

```
bitnami-gitlab-ce-15-15.9.3-ce.0-r0-debian-11-amd64 [Работаer] - Oracle VM VirtualBox
Файл  Машина  Вид  Ввод  Устройства  Справка
* runit_service[nginx] action restart (up to date)
[2023-03-13T01:08:41+00:00] INFO: Cinc Client Run complete in 98.929926527 seconds

Running handlers:
[2023-03-13T01:08:41+00:00] INFO: Running report handlers
Running handlers complete
[2023-03-13T01:08:42+00:00] INFO: Report handlers complete
Infra Phase complete, 6/795 resources updated in 01 minutes 40 seconds
gitlab Reconfigured!
bitnami@debian:~/sample-gitlabci-cpp-project-master$ sudo gitlab-runner register --url http://192.168.0.168/ --registration-token GR1348941sup9RYAV4yhFLwGMnc9M
Runtime platform                          arch=amd64 os=linux pid=10505 revision=d540b510
version=15.9.1
Running in system-mode.

Enter the GitLab instance URL (for example, https://gitlab.com/):
http://192.168.0.168/1:
Enter the registration token:
GR1348941sup9RYAV4yhFLwGMnc9M1:
Enter a description for the runner:
[debian1]:
Enter tags for the runner (comma-separated):

Enter optional maintenance note for the runner:

WARNING: Support for registration tokens and runner parameters in the 'register' command has been deprecated in GitLab Runner 15.6 and will be replaced with support for authentication tokens. For more information, see https://gitlab.com/gitlab-org/gitlab/-/issues/380872
Registering runner... succeeded runner=GR1348941sup9RYAV
Enter an executor: custom, docker, docker-ssh, virtualbox, parallels, shell, ssh, docker+machine, docker-ssh+machine, instance, kubernetes:
shell
Runner registered successfully. Feel free to start it, but if it's running already the config should be automatically reloaded!

Configuration (with the authentication token) was saved in "/etc/gitlab-runner/config.toml"
bitnami@debian:~/sample-gitlabci-cpp-project-master$ _
```


Step 4

Changes 1

Pipelines 1

Showing 1 changed file with 5 additions and 16 deletions

Hide whitespace changes

Inline

Side-by-side

▼ .gitlab-ci.yml

+5 -16

View file @3fa3daa2

```
1 - # Specify the docker image to use (only used if using docker runners)
2 - # See: http://doc.gitlab.com/ee/ci/docker/using_docker_images.html
3 - image: ubuntu:14.04
4 -
5 - # Define commands that run before each job's script
6 1   before_script:
7 2     - apt-get update
8 3     - apt-get install -y gcc g++
9 4
10 - # Try to compile our sample hello world app
11 - compile:
12 -   script:
13 -     # Compile our app
14 -     - g++ helloworld.cpp -o helloworld
15 -     # Verify that our compiled app works properly with a custom "test" script
16 -     - ./verify.sh
17 -     # Save the compiled output from the above for downloading via GitLab and in GitLab 8.6 to use in future
18 -     build steps
19 -   artifacts:
20 -     paths:
21 -     - helloworld
22
23 + job:
24 +   script:
25 +     - g++ helloworld.cpp -o helloworld
26 +     - chmod 777 verify.sh
27 +     - ./verify.sh
```

Step 5

The screenshot displays the GitHub Actions workflow runner interface. The top navigation bar shows the repository name 'sample-github-cpp-project'. On the left sidebar, the 'Jobs' tab is selected. The main panel shows a job named 'job' triggered by Administrator, which has passed. Below this, a search bar for the job log is present. The job log itself is displayed as a scrollable area with a dark background and light-colored text, showing a series of steps from 1 to 24. Steps 1 through 10 are expanded, revealing their respective commands. The final output of step 24 is 'Job succeeded'. On the right side of the interface, there is a summary section for the job, indicating it took 1 second to complete. Below this, there is a link to view the commit history and a dropdown menu for selecting a pipeline.

```

1 Running with gitlab-runner 15.9.1 (6d6db33f)
2 on debian @riscv64 system ID: s_1628dcirc919
3 Preparing the "shell" executor
4 Using Shell Docker executor...
5 Preparing dockerimages
6
7 Running on debian...
8 Setting source from git repository
9 Fetching changes with git depth set to 20...
10 Initializing existing git repository in /home/gitlab-runner/builds/@riscv64/runt/sample-github-cpp-project/.git/
11 Checking out 0f1d3d64 as detached HEAD (ref is master)...
12 Skipping git submodule setup
13 Executing "run_script" stage of the job script
14 $ g++ helloWorld.cpp -o HelloWorld
15 $ ./HelloWorld
16 $ ./verify.sh
17 Starting sample CI verification script
18 Trying to execute ./helloWorld
19 Setup is OK
20 Output is correct, OK
21 Job succeeded

```