

# Gitlab

## Docker Compose

- `mkdir -p /data/docker/gitlab/gitlab /data/docker/gitlab/redis /data/docker/gitlab/postgresql`
- `chown -R 777 /data/docker/gitlab/gitlab /data/docker/gitlab/redis /data/docker/gitlab/postgresql`
- `docker-compose` `docker-compose.yml`

```
gitlab:
  image: sameersbn/gitlab:10.4.2-1
  ports:
    - "10022:22"
    - "10080:80"
  links:
    - gitlab-redis:redisio
    - gitlab-postgresql:postgresql
  environment:
    - GITLAB_PORT=80
    - GITLAB_SSH_PORT=22
    - GITLAB_SECRETS_DB_KEY_BASE=long-and-random-alpha-numeric-string
    - GITLAB_SECRETS_SECRET_KEY_BASE=long-and-random-alpha-numeric-string
    - GITLAB_SECRETS_OTP_KEY_BASE=long-and-random-alpha-numeric-string
  volumes:
    - /data/docker/gitlab/gitlab:/home/git/data
  restart: always
gitlab-redis:
  image: sameersbn/redis
  volumes:
    - /data/docker/gitlab/redis:/var/lib/redis
  restart: always
gitlab-postgresql:
  image: sameersbn/postgresql:9.6-2
  environment:
    - DB_NAME=gitlabhq_production
    - DB_USER=gitlab
    - DB_PASS=password
    - DB_EXTENSION=pg_trgm
```

```
volumes:
  - /data/docker/gitlab/postgresql: /var/lib/postgresql

restart: always
```

- `docker-compose up -d`
- <http://192.168.0.105:10080>

# Gitlab High Availability

- <https://about.gitlab.com/high-availability/>
- Gitlab nginx
- 4G
- <https://about.gitlab.com/products/>
- <https://about.gitlab.com/>
- <https://www.gitlab.com.cn/>
- <https://about.gitlab.com/downloads/>
- <https://about.gitlab.com/installation/#centos-7>
- - <https://mirror.tuna.tsinghua.edu.cn/help/gitlab-ce/>
- <https://ken.io/note/centos7-gitlab-install-tutorial>

```
sudo yum install -y curl policycoreutils-python openssh-server
```

```
sudo systemctl enable sshd
```

```
sudo systemctl start sshd
```

```
curl https://packages.gitlab.com/install/repositories/gitlab/gitlab-ce/script.rpm.sh | sudo
bash
```

```
sudo EXTERNAL_URL="http://192.168.1.123:8181" yum install -y gitlab-ce
```

- `/ IP`
  - `sudo vim /etc/gitlab/gitlab.rb`
  - `external_url 'http://gitlab.example.com'` / IP
  - `sudo gitlab-ctl reconfigure`
  - `sudo gitlab-ctl start`
  - `sudo gitlab-ctl stop`
  - `sudo gitlab-ctl restart`
- `http://192.168.1.111:80`

- `root` `root` `gitlab123456` 8
- root RESTful
  - <http://192.168.1.111/admin/users>
  - <http://192.168.1.111/admin/groups>
  - <http://192.168.1.111/admin/projects>
  - SSH Key <http://192.168.1.111/profile/keys>
  - <http://192.168.1.111/admin/users/> /edit
- - 
  - SSH Key <http://192.168.1.111/profile/keys>

# Jenkins

- Gitlab Access Token [http://192.168.0.105:10080/profile/personal\\_access\\_tokens](http://192.168.0.105:10080/profile/personal_access_tokens)
  - Name
  - `Access the authenticated user's API`
  - Create personal access token `wt93jQzA8yu5a6pfsk3s` Jenkinsfile
- Jenkins <http://192.168.0.105:18080/pluginManager/available>
  - Gitlab hpi
  - Gitlab Hook GitLab WebHooks
  - Gitlab Authentication GitLab
- Jenkins Jenkins-->Credentials-->System-->Global credentials(unrestricted)-->Add Credentials
  - [http://192.168.0.105:18080/credentials/store/system/domain/\\_/newCredentials](http://192.168.0.105:18080/credentials/store/system/domain/_/newCredentials)
  - kind `GitLab API token`
  - token access token
  - ID Gitlab
- <http://192.168.1.111/help/user/permissions>
- `Guest Reporter Developer Master Owner`
- `Add user(s) to the group` <http://192.168.1.111/admin/groups/>

	Guest	Reporter	Developer	Master	Owner
	✓	✓	✓	✓	✓
					✓
				✓	✓
					✓
					✓

- Guest Reporter Developer Master Owner
  - Guest
  - Reporter ; issue
  - Developer ;
  - Master ; Master
  - Owner ;
- Members <http://192.168.1.111/> / </settings/members>

	Guest	Reporter	Developer	Master	Owner
issue	✓	✓	✓	✓	✓
	✓	✓	✓	✓	✓
		✓	✓	✓	✓
		✓	✓	✓	✓
		✓	✓	✓	✓
			✓	✓	✓
			✓	✓	✓
			✓	✓	✓
			✓	✓	✓
			✓	✓	✓
tag			✓	✓	✓
wiki			✓	✓	✓
issue			✓	✓	✓
labels			✓	✓	✓
				✓	✓
				✓	✓
				✓	✓
				✓	✓
/ tag				✓	✓

	Guest	Reporter	Developer	Master	Owner
				✓	✓
deploy keys				✓	✓
hooks				✓	✓
visibility level					✓
namespace					✓
					✓
					✓
					✓

- <http://192.168.1.111/ / /settings/members>
  - Import [http://192.168.1.111/ / /project\\_members/import](http://192.168.1.111/ / /project_members/import)

## merge

- <http://192.168.1.111/ / /settings/repository#>
- CI key

# Gitlab

- <http://119.23.252.150:10080/admin/users/>
- 
- <http://119.23.252.150:10080/groups>
- Visibility Level
  - Private The group and its projects can only be viewed by members.
  - Internal Gitlab The group and any internal projects can be viewed by any logged user.
  - Public The group and any public projects can be viewed without any authentication.
- <http://119.23.252.150:10080/>

## SSH keys

- <http://119.23.252.150:10080/profile/keys>
- <http://119.23.252.150:10080/help/ssh/README>
- SSH key `ssh-keygen -t rsa -C "gitnavi@qq.com" -b 4096`

- linux    SSH key `cat ~/.ssh/id_rsa.pub`    gitlab

# Gitlab - Git flow

- Git flow    Git

- - <http://nvie.com/posts/a-successful-git-branching-model/>
    - Git-flow <https://github.com/nvie/gitflow>
    - Git-flow
      - <https://www.git-tower.com/learn/git/ebook/cn/command-line/advanced-topics/git-flow>
      - [https://danielkummer.github.io/git-flow-cheatsheet/index.zh\\_CN.html](https://danielkummer.github.io/git-flow-cheatsheet/index.zh_CN.html)
  - <http://www.ruanyifeng.com/blog/2015/12/git-workflow.html>
  - <https://zhangmengpl.gitbooks.io/gitlab-guide/content/whatisgitflow.html>
  - <http://blog.jobbole.com/76867/>
  - <http://www.cnblogs.com/cnblogsfans/p/5075073.html>
- - <https://docs.gitlab.com/ce/integration/omniauth.html>
  - [https://docs.gitlab.com/ce/integration/oauth2\\_generic.html](https://docs.gitlab.com/ce/integration/oauth2_generic.html)
  - <https://gitlab.com/satorix/omniauth-oauth2-generic#gitlab-config-example>
- gitlab    gitlab
- gitlab    clone    gitlab    ssh-key
- block\_auto\_created\_users=true    false
- `|sudo vim /etc/gitlab/gitlab.rb|`    309
  - oauth2\_generic    gem

```
gitlab_rails['omniauth_enabled'] = true
gitlab_rails['omniauth_allow_single_sign_on'] = ['google_oauth2', 'facebook', 'twitter',
'oauth2_generic']
gitlab_rails['omniauth_block_auto_created_users'] = false
gitlab_rails['omniauth_sync_profile_attributes'] = ['email', 'username']
gitlab_rails['omniauth_external_providers'] = ['google_oauth2', 'facebook', 'twitter',
'oauth2_generic']
gitlab_rails['omniauth_providers'] = [
  {
    "name" => "google_oauth2",
    "label" => "Google",
    "app_id" => "123456",
```

```

    "app_secret" => "123456",
    "args" => {
        "access_type" => 'offline',
        "approval_prompt" => '123456'
    }
},
{
    "name" => "facebook",
    "label" => "facebook",
    "app_id" => "123456",
    "app_secret" => "123456"
},
{
    "name" => "twitter",
    "label" => "twitter",
    "app_id" => "123456",
    "app_secret" => "123456"
},
{
    "name" => "oauth2_generic",
    "app_id" => "123456",
    "app_secret" => "123456",
    "args" => {
        client_options: {
            "site" => "http://sso.cdk8s.com:9090/sso",
            "user_info_url" => "/oauth/userinfo"
        },
        user_response_structure: {
            root_path: ["user_attribute"],
            attributes: {
                "nickname": "username"
            }
        }
    }
}
]

```

- <http://blog.smallmuou.xyz/git/2016/03/11/Gitlab-GitFlow.html>
- <https://zhangmengpl.gitbooks.io/gitlab-guide/content/whatisgitflow.html>

- <https://blog.coderstory.cn/2017/02/01/gitlab/>
  - <https://xuanwo.org/2016/04/13/gitlab-install-intro/>
  - <https://softlms.github.io/2016/11/14/jenkins-gitlab-docker/>
- 

Revision #2

Created 20 March 2020 15:27:23 by

Updated 14 August 2020 06:18:41 by