

# 第三周作业

1.基于docker-compose实现对nginx+tomcat web服务的单机编排（第二周已做） 2.安装gitlab、创建group、user和project并授权 3.熟练git命令的基本使用，通过git命令实现源代码的clone、push等基本操作 4.熟练掌握对gitlab服务的数据备份与恢复 5.部署jenkins服务器并安装gitlab插件、实现代码免秘钥代码clone

## 1. 安装gitlab、创建group、user和project并授权

### 1.1. 安装

#### 1.1.1. 安装报错

```
1 root@gitlab-01:~# dpkg -i gitlab-ce_15.4.3-ce.0_amd64.deb
2 Selecting previously unselected package gitlab-ce.
3 (Reading database ... 66366 files and directories currently installed.)
4 Preparing to unpack gitlab-ce_15.4.3-ce.0_amd64.deb ...
5 Unpacking gitlab-ce (15.4.3-ce.0) ...
6 Setting up gitlab-ce (15.4.3-ce.0) ...
7 [2022-11-12T07:00:19+00:00] WARN: Please install an English UTF-8 locale for Chef
8 Infra Client to use, falling back to C locale and disabling UTF-8 support.
9 It looks like there was a problem with public attributes; run gitlab-ctl reconfigure
10 manually to fix.
11 dpkg: error processing package gitlab-ce (--install):
12 installed gitlab-ce package post-installation script subprocess returned error
13 status 1
14 Errors were encountered while processing:
15 gitlab-ce
16 root@gitlab-01:~#
17 #报错由于没有en语言包，需要安装。
18 #查看当前语言包
19 root@gitlab-01:~# locale -a
20 C
21 C.utf8
22 POSIX
23 zh_CN.utf8
24 zh_HK.utf8
25 zh_SG.utf8
26 zh_TW.utf8
27 #卸载并中文包并安装en语言包
28 root@gitlab-01:~# apt-get purge locales
29 root@gitlab-01:~# apt-get purge language-pack-en
30 root@gitlab-01:~# apt-get install locales
31 root@gitlab-01:~# apt-get install language-pack-en
32 root@gitlab-01:~# locale -a
33 C
34 C.utf8
35 POSIX
36 en_AG
```

```

35 en_AG.utf8
36 en_AU.utf8
37 en_BW.utf8
38 en_CA.utf8
39 en_DK.utf8
40 en_GB.utf8
41 en_HK.utf8
42 en_IE.utf8
43 en_IL
44 en_IL.utf8
45 en_IN
46 en_IN.utf8
47 en_NG
48 en_NG.utf8
49 en_NZ.utf8
50 en_PH.utf8
51 en_SG.utf8
52 en_US.utf8
53 en_ZA.utf8
54 en_ZM
55 en_ZM.utf8
56 en_ZW.utf8

```

### 1.1.2. 执行安装gitlab

```

1 root@gitlab-01:~# dpkg -i gitlab-ce_15.4.3-ce.0_amd64.deb
2 Selecting previously unselected package gitlab-ce.
3 (Reading database ... 66480 files and directories currently installed.)
4 Preparing to unpack gitlab-ce_15.4.3-ce.0_amd64.deb ...
5 Unpacking gitlab-ce (15.4.3-ce.0) ...
6 Setting up gitlab-ce (15.4.3-ce.0) ...
7 It looks like GitLab has not been configured yet; skipping the upgrade script.
8
9      *.              *.
10     ***             ***
11     *****        *****
12     .*****        *****
13     *****        *****
14     ////////////// //*****// //////////////
15     ////////////// //*****// //////////////
16     .////////// //*****// //////////////
17     ////////////// //*****// //////////////
18     ////////////// //*****// //////////////
19     ./// //*****// //////////////
20     //*****//
21
22
23
24
25     /  _  _  ( )  /  _  _  /  _  _  /  _  _  /
26     /  /  _  /  /  _  /  /  _  _  /  _  _  \
27     /  /  /  /  /  /  /  _  /  /  /  /  /  /
28     \  _  /  \  _  /  _  /  \  _  /  \  _  /

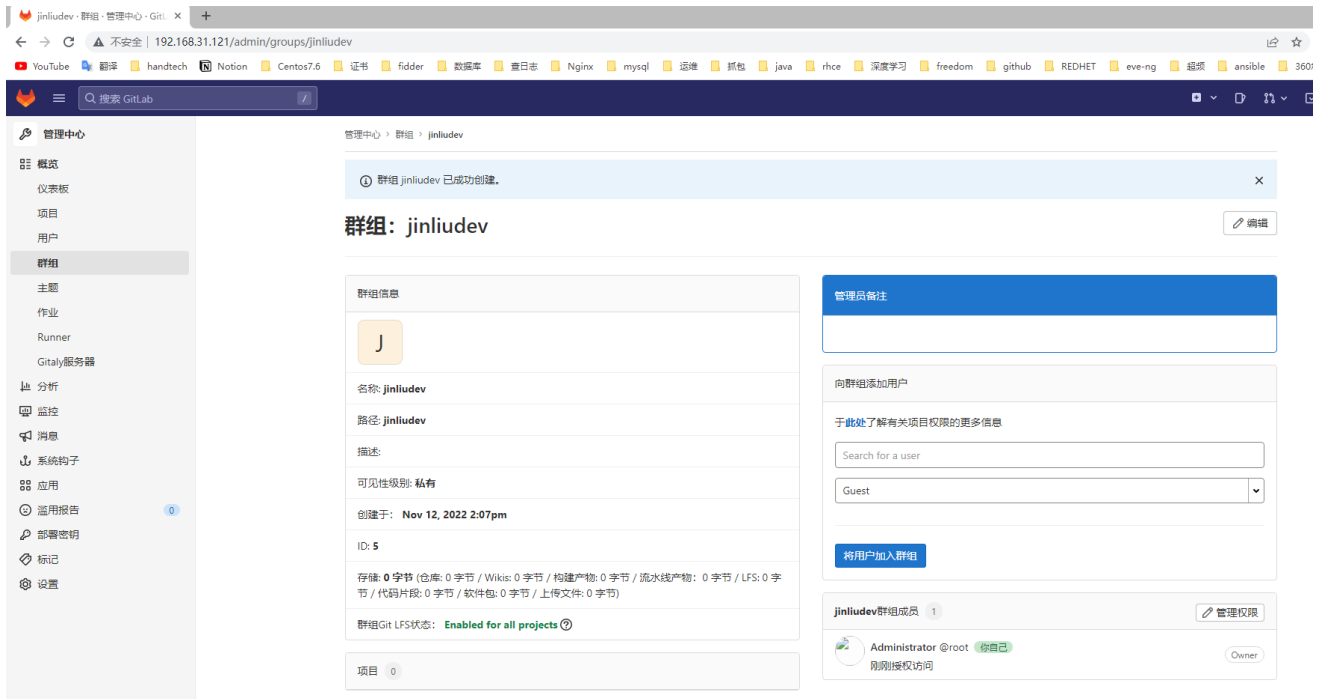
```

```
29
30
31 Thank you for installing GitLab!
32 GitLab was unable to detect a valid hostname for your instance.
33 Please configure a URL for your GitLab instance by setting `external_url`
34 configuration in /etc/gitlab/gitlab.rb file.
35 Then, you can start your GitLab instance by running the following command:
36     sudo gitlab-ctl reconfigure
37
38 For a comprehensive list of configuration options please see the Omnibus GitLab
39 readme
40 https://gitlab.com/gitlab-org/omnibus-gitlab/blob/master/README.md
41
42 Help us improve the installation experience, let us know how we did with a 1 minute
43 survey:
44 https://gitlab.fra1.qualtrics.com/jfe/form/SV_6kVqZANThUQ1bZb?
45 installation=omnibus&release=15-4
```

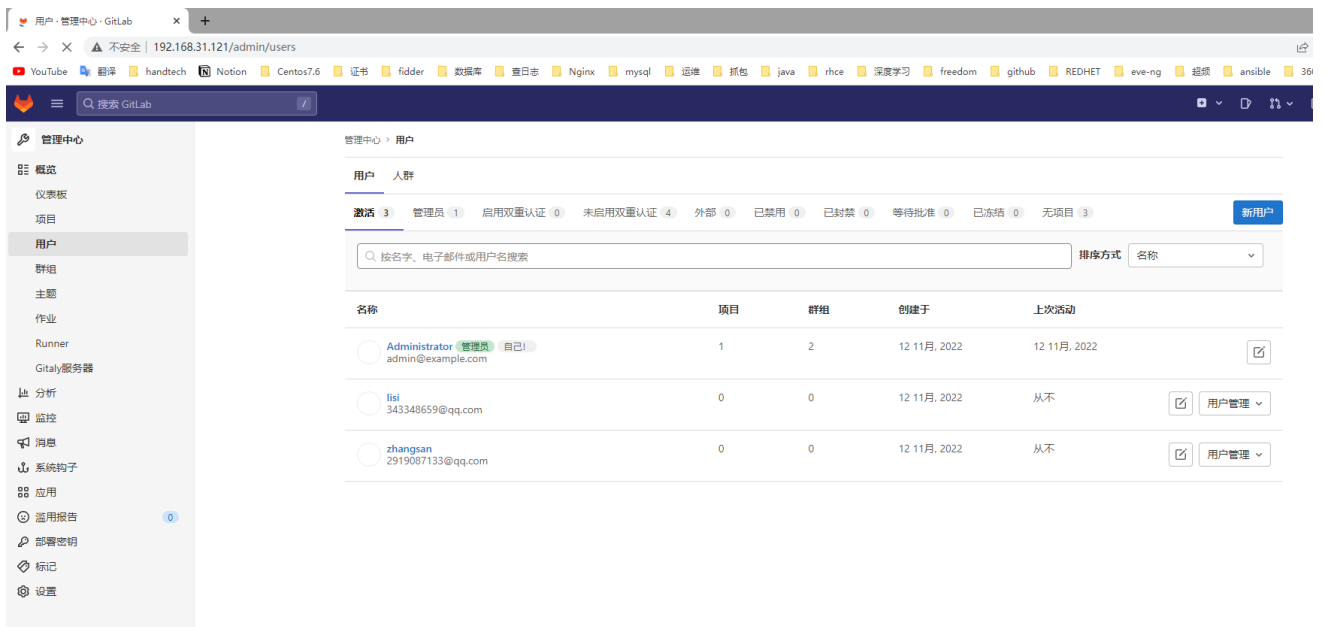
### 1.1.3. 修改配置

```
1  external_url 'http://192.168.31.121'
2  gitlab_rails['smtp_enable'] = true
3  gitlab_rails['smtp_address'] = "smtp.qq.com"
4  gitlab_rails['smtp_port'] = 465
5  gitlab_rails['smtp_user_name'] = "1564229817@qq.com"
6  gitlab_rails['smtp_password'] = "*****"
7  gitlab_rails['smtp_domain'] = "qq.com"
8  gitlab_rails['smtp_authentication'] = "login"
9  gitlab_rails['smtp_enable_starttls_auto'] = true
10 gitlab_rails['smtp_tls'] = true
11 gitlab_rails['gitlab_email_from'] = '1564229817@qq.com'
12 user['git_user_email'] = "1564229817@qq.com"
```

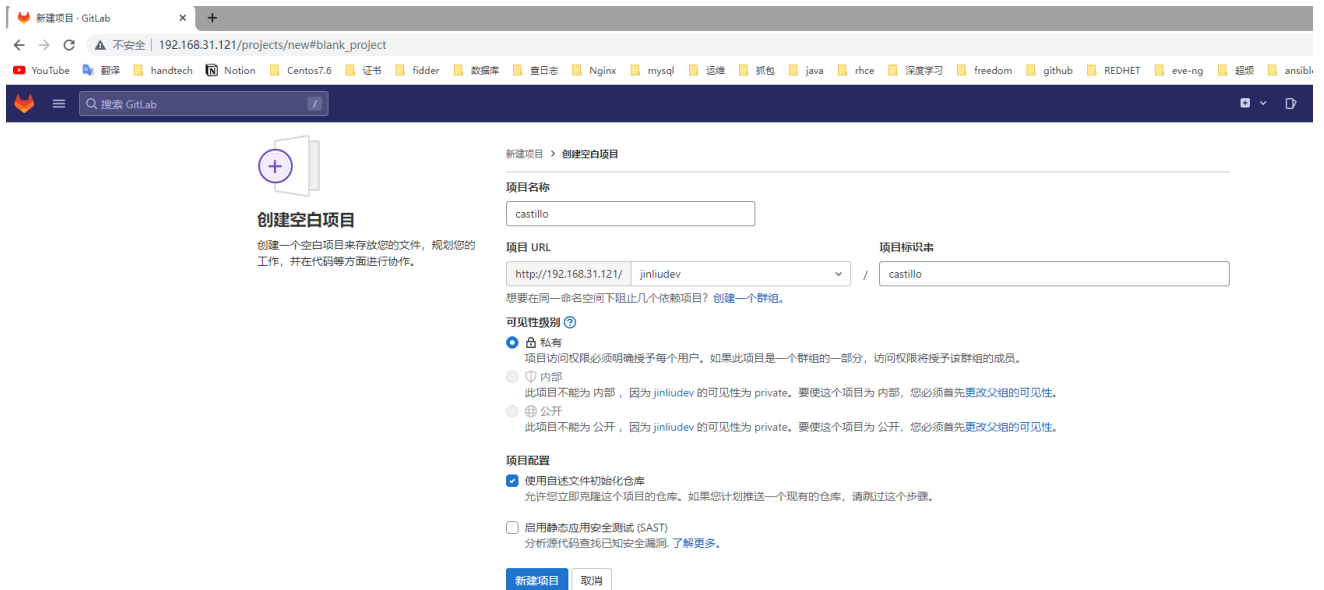
## 1.2. 创建group



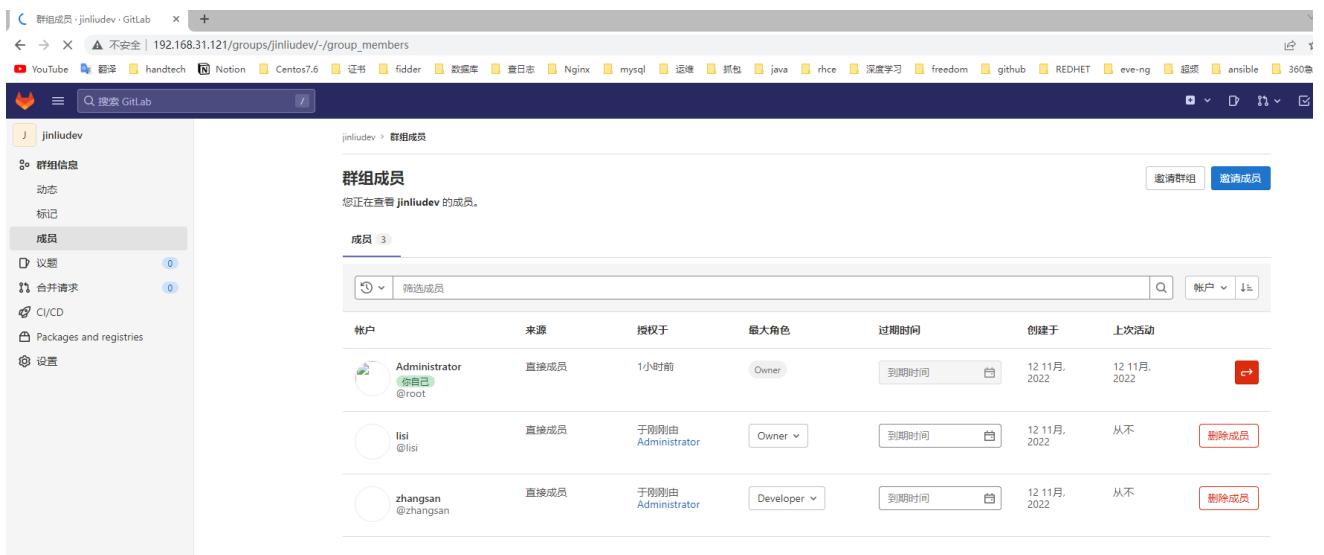
## 1.3. 创建用户



## 1.4. 创建项目



## 1.5. 授权



## 2. 熟练git命令的基本使用，通过git命令实现源代码的clone、push等基本操作

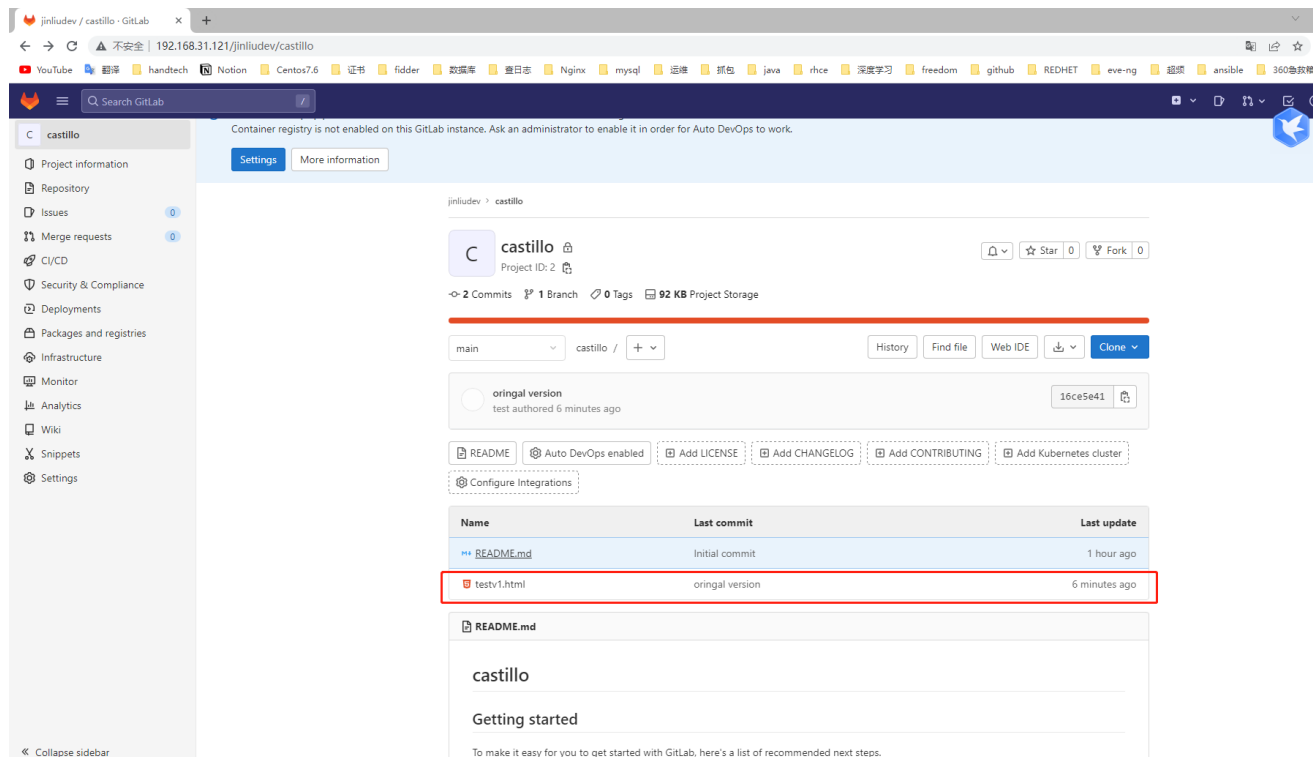
### 2.1. clone&push

```
1 #在开发机clone项目
2 castillo@DESKTOP-FAPNCRD MINGW64 /c/devtest
3 $ git clone http://192.168.31.121/jinliudev/castillo.git
4 Cloning into 'castillo'...
5 remote: Enumerating objects: 3, done.
6 remote: Counting objects: 100% (3/3), done.
7 remote: Compressing objects: 100% (2/2), done.
8 remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
9 Receiving objects: 100% (3/3), done.
```

10

```
11 castillo@DESKTOP-FAPNCRD MINGW64 /c/devtest
12 $ cd castillo/
13
14 #将代码上传
15 castillo@DESKTOP-FAPNCRD MINGW64 /c/devtest/castillo (main)
16 $ git add .
17
18 castillo@DESKTOP-FAPNCRD MINGW64 /c/devtest/castillo (main)
19 #设置备注
20 $ git commit -m "oringal version"
21 Author identity unknown
22
23 *** Please tell me who you are.
24
25 Run
26
27     git config --global user.email "you@example.com"
28     git config --global user.name "Your Name"
29
30 to set your account's default identity.
31 Omit --global to set the identity only in this repository.
32
33 fatal: unable to auto-detect email address (got 'castillo@DESKTOP-FAPNCRD.(none)')
34
35 castillo@DESKTOP-FAPNCRD MINGW64 /c/devtest/castillo (main)
36 $ git config --global user.email "test@qq.com"
37
38 castillo@DESKTOP-FAPNCRD MINGW64 /c/devtest/castillo (main)
39 $ git config --global user.name "test"
40
41 castillo@DESKTOP-FAPNCRD MINGW64 /c/devtest/castillo (main)
42 $ git commit -m "oringal version"
43 [main 16ce5e4] oringal version
44 1 file changed, 10 insertions(+)
45 create mode 100644 testv1.html
46
47 castillo@DESKTOP-FAPNCRD MINGW64 /c/devtest/castillo (main)
48
49 #推送代码
50 $ git push
51 Enumerating objects: 4, done.
52 Counting objects: 100% (4/4), done.
53 Delta compression using up to 2 threads
54 Compressing objects: 100% (3/3), done.
55 Writing objects: 100% (3/3), 389 bytes | 389.00 KiB/s, done.
56 Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
57 To http://192.168.31.121/jinliudev/castillo.git
58    7526e1f..16ce5e4  main -> main
59
60 castillo@DESKTOP-FAPNCRD MINGW64 /c/devtest/castillo (main)
61 $
62
```

## 2.2. 验证结果



## 3. 熟练掌握对gitlab服务的数据备份与恢复

### 3.1. 简单回滚

```
1 castillo@DESKTOP-FAPNCRD MINGW64 /c/devtest/castillo (main)
2 $ git status
3 On branch main
4 Your branch is behind 'origin/main' by 1 commit, and can be fast-forwarded.
5   (use "git pull" to update your local branch)
6
7 nothing to commit, working tree clean
8
9 castillo@DESKTOP-FAPNCRD MINGW64 /c/devtest/castillo (main)
10 $ git pull
11 Updating 7526elf..16ce5e4
12 Fast-forward
13  testv1.html | 10 ++++++++
14  1 file changed, 10 insertions(+)
15  create mode 100644 testv1.html
16
17 castillo@DESKTOP-FAPNCRD MINGW64 /c/devtest/castillo (main)
18 $ git add .
19
20 castillo@DESKTOP-FAPNCRD MINGW64 /c/devtest/castillo (main)
21 $ git commit -m "v2"
22 [main 3601a17] v2
23  1 file changed, 1 insertion(+)
24
25 castillo@DESKTOP-FAPNCRD MINGW64 /c/devtest/castillo (main)
```

```

26 $ git push
27 Enumerating objects: 5, done.
28 Counting objects: 100% (5/5), done.
29 Delta compression using up to 2 threads
30 Compressing objects: 100% (3/3), done.
31 Writing objects: 100% (3/3), 328 bytes | 328.00 KiB/s, done.
32 Total 3 (delta 1), reused 0 (delta 0), pack-reused 0
33 To http://192.168.31.121/jinliudev/castillo.git
34    16ce5e4..3601a17  main -> main
35
36 castillo@DESKTOP-FAPNCRD MINGW64 /c/devtest/castillo (main)
37 $ git reset --hard HEAD^
38 HEAD is now at 16ce5e4 oringal version
39
40 castillo@DESKTOP-FAPNCRD MINGW64 /c/devtest/castillo (main)
41 $
42

```

## 3.2. 关键命令

```

1  #git reset --hard HEAD^^ #git版本回滚, HEAD为当前版本, 加一个^为上一个, ^^为上上一个版本
2  #git reflog #获取每次提交的ID, 可以使用--hard根据提交的ID进行版本回退
3  castillo@DESKTOP-FAPNCRD MINGW64 /c/devtest/castillo (main)
4  $ git reflog
5  16ce5e4 (HEAD -> main) HEAD@{0}: reset: moving to HEAD^
6  3601a17 (origin/main, origin/HEAD) HEAD@{1}: commit: v2
7  16ce5e4 (HEAD -> main) HEAD@{2}: pull: Fast-forward
8  7526elf HEAD@{3}: reset: moving to HEAD^
9  16ce5e4 (HEAD -> main) HEAD@{4}: commit: oringal version
10 7526elf HEAD@{5}: clone: from http://192.168.31.121/jinliudev/castillo.git
11
12 castillo@DESKTOP-FAPNCRD MINGW64 /c/devtest/castillo (main)
13 $
14 #git reset --hard <提交ID> #回退到指定id的版本

```

## 3.3. 其它命令

```

1  git config --global user.name "NAME" #设置全局用户名
2  git config --global user.email xxxx@yy.com #设置全局邮箱
3  git config --global --list #列出用户全局设置
4  # git branch #查看当前所处的分支
5  #git checkout -b develop #创建并切换到一个新分支
6  #git checkout develop #切换分支
7  git log #查看操作日志
8  vim .gitignore #定义忽略文件上传至gitlab
9
10 git clone #克隆代码到本地
11 git push #提交代码到服务器
12 git pull #更新本地代码
13 git add index.html、abc/、. #添加指定文件、目录或当前目录下所有数据到暂存区
14 git commit -m "xx" #提交文件到工作区

```



## 3.4. gitlab数据备份与恢复

### 3.4.1. 备份

```
1 root@gitlab-01:~# gitlab-ctl stop unicorn sidekiq
2 root@gitlab-01:~# mkdir /opt/gitbackup
3 root@gitlab-01:~# cd /opt/git
4 gitbackup/ gitlab/
5 root@gitlab-01:~# cd /opt/gitbackup/
6 root@gitlab-01:/opt/gitbackup# gitlab-rake gitlab:backup:create
7 2022-11-12 17:30:40 +0000 -- Dumping main_database ...
8 Dumping PostgreSQL database gitlabhq_production ... [DONE]
9 2022-11-12 17:30:43 +0000 -- Dumping main_database ... done
10 2022-11-12 17:30:43 +0000 -- Dumping ci_database ... [DISABLED]
11 2022-11-12 17:30:43 +0000 -- Dumping repositories ...
12 {"command":"create","gl_project_path":"gitlab-instance-
b269664d/Monitoring","level":"info","msg":"started
create","relative_path":"@hashed/6b/86/6b86b273ff34fcel9d6b804eff5a3f5747ada4eaa22f1d
49c01e52ddb7875b4b.git","storage_name":"default","time":"2022-11-12T17:30:44.757Z"}
13 {"command":"create","gl_project_path":"gitlab-instance-
b269664d/Monitoring.wiki","level":"info","msg":"started
create","relative_path":"@hashed/6b/86/6b86b273ff34fcel9d6b804eff5a3f5747ada4eaa22f1d
49c01e52ddb7875b4b.wiki.git","storage_name":"default","time":"2022-11-
12T17:30:44.757Z"}
14 {"command":"create","error":"manager: repository empty: repository
skipped","gl_project_path":"gitlab-instance-
b269664d/Monitoring.wiki","level":"warning","msg":"skipped
create","relative_path":"@hashed/6b/86/6b86b273ff34fcel9d6b804eff5a3f5747ada4eaa22f1d
49c01e52ddb7875b4b.wiki.git","storage_name":"default","time":"2022-11-
12T17:30:45.815Z"}
15 {"command":"create","gl_project_path":"gitlab-instance-
b269664d/Monitoring","level":"info","msg":"started
create","relative_path":"@hashed/6b/86/6b86b273ff34fcel9d6b804eff5a3f5747ada4eaa22f1d
49c01e52ddb7875b4b.design.git","storage_name":"default","time":"2022-11-
12T17:30:45.816Z"}
16 {"command":"create","error":"manager: repository empty: repository
skipped","gl_project_path":"gitlab-instance-
b269664d/Monitoring","level":"warning","msg":"skipped
create","relative_path":"@hashed/6b/86/6b86b273ff34fcel9d6b804eff5a3f5747ada4eaa22f1d
49c01e52ddb7875b4b.git","storage_name":"default","time":"2022-11-12T17:30:45.833Z"}
17 {"command":"create","gl_project_path":"jinliudev/castillo","level":"info","msg":"star
ted
create","relative_path":"@hashed/d4/73/d4735e3a265e16eee03f59718b9b5d03019c07d8b6c51f
90da3a666eec13ab35.git","storage_name":"default","time":"2022-11-12T17:30:45.833Z"}

18 {"command":"create","error":"manager: repository empty: repository
```

```
    skipped", "gl_project_path": "gitlab-instance-
b269664d/Monitoring", "level": "warning", "msg": "skipped
create", "relative_path": "@hashed/6b/86/6b86b273ff34fce19d6b804eff5a3f5747ada4eaa22f1d
49c01e52ddb7875b4b.design.git", "storage_name": "default", "time": "2022-11-
12T17:30:45.839Z"}
19 {"command": "create", "gl_project_path": "jinliudev/castillo.wiki", "level": "info", "msg":
"started
create", "relative_path": "@hashed/d4/73/d4735e3a265e16eee03f59718b9b5d03019c07d8b6c51f
90da3a666eec13ab35.wiki.git", "storage_name": "default", "time": "2022-11-
12T17:30:45.839Z"}
20 {"command": "create", "error": "manager: repository empty: repository
skipped", "gl_project_path": "jinliudev/castillo.wiki", "level": "warning", "msg": "skipped
create", "relative_path": "@hashed/d4/73/d4735e3a265e16eee03f59718b9b5d03019c07d8b6c51f
90da3a666eec13ab35.wiki.git", "storage_name": "default", "time": "2022-11-
12T17:30:45.926Z"}
21 {"command": "create", "gl_project_path": "jinliudev/castillo", "level": "info", "msg": "star
ted
create", "relative_path": "@hashed/d4/73/d4735e3a265e16eee03f59718b9b5d03019c07d8b6c51f
90da3a666eec13ab35.design.git", "storage_name": "default", "time": "2022-11-
12T17:30:45.926Z"}
22 {"command": "create", "error": "manager: repository empty: repository
skipped", "gl_project_path": "jinliudev/castillo", "level": "warning", "msg": "skipped
create", "relative_path": "@hashed/d4/73/d4735e3a265e16eee03f59718b9b5d03019c07d8b6c51f
90da3a666eec13ab35.design.git", "storage_name": "default", "time": "2022-11-
12T17:30:45.927Z"}
23 {"command": "create", "gl_project_path": "jinliudev/castillo", "level": "info", "msg": "comp
leted
create", "relative_path": "@hashed/d4/73/d4735e3a265e16eee03f59718b9b5d03019c07d8b6c51f
90da3a666eec13ab35.git", "storage_name": "default", "time": "2022-11-12T17:30:46.149Z"}
24 2022-11-12 17:30:46 +0000 -- Dumping repositories ... done
25 2022-11-12 17:30:46 +0000 -- Dumping uploads ...
26 2022-11-12 17:30:46 +0000 -- Dumping uploads ... done
27 2022-11-12 17:30:46 +0000 -- Dumping builds ...
28 2022-11-12 17:30:46 +0000 -- Dumping builds ... done
29 2022-11-12 17:30:46 +0000 -- Dumping artifacts ...
30 2022-11-12 17:30:46 +0000 -- Dumping artifacts ... done
31 2022-11-12 17:30:46 +0000 -- Dumping pages ...
32 2022-11-12 17:30:46 +0000 -- Dumping pages ... done
33 2022-11-12 17:30:46 +0000 -- Dumping lfs objects ...
34 2022-11-12 17:30:46 +0000 -- Dumping lfs objects ... done
35 2022-11-12 17:30:46 +0000 -- Dumping terraform states ...
36 2022-11-12 17:30:46 +0000 -- Dumping terraform states ... done
37 2022-11-12 17:30:46 +0000 -- Dumping container registry images ... [DISABLED]
38 2022-11-12 17:30:46 +0000 -- Dumping packages ...
39 2022-11-12 17:30:46 +0000 -- Dumping packages ... done
40 2022-11-12 17:30:46 +0000 -- Creating backup archive:
1668274240_2022_11_12_15.4.3_gitlab_backup.tar ...
41 2022-11-12 17:30:46 +0000 -- Creating backup archive:
1668274240_2022_11_12_15.4.3_gitlab_backup.tar ... done
42 2022-11-12 17:30:46 +0000 -- Uploading backup archive to remote storage ...
[SKIPPED]
43 2022-11-12 17:30:46 +0000 -- Deleting old backups ... [SKIPPED]
44 2022-11-12 17:30:46 +0000 -- Deleting tar staging files ...
```

```
45 2022-11-12 17:30:46 +0000 -- Cleaning up
   /var/opt/gitlab/backups/backup_information.yml
46 2022-11-12 17:30:46 +0000 -- Cleaning up /var/opt/gitlab/backups/db
47 2022-11-12 17:30:46 +0000 -- Cleaning up /var/opt/gitlab/backups/repositories
48 2022-11-12 17:30:46 +0000 -- Cleaning up /var/opt/gitlab/backups/uploads.tar.gz
49 2022-11-12 17:30:46 +0000 -- Cleaning up /var/opt/gitlab/backups/builds.tar.gz
50 2022-11-12 17:30:46 +0000 -- Cleaning up /var/opt/gitlab/backups/artifacts.tar.gz
51 2022-11-12 17:30:46 +0000 -- Cleaning up /var/opt/gitlab/backups/pages.tar.gz
52 2022-11-12 17:30:46 +0000 -- Cleaning up /var/opt/gitlab/backups/lfs.tar.gz
53 2022-11-12 17:30:46 +0000 -- Cleaning up
   /var/opt/gitlab/backups/terraform_state.tar.gz
54 2022-11-12 17:30:46 +0000 -- Cleaning up /var/opt/gitlab/backups/packages.tar.gz
55 2022-11-12 17:30:46 +0000 -- Deleting tar staging files ... done
56 2022-11-12 17:30:46 +0000 -- Deleting backups/tmp ...
57 2022-11-12 17:30:46 +0000 -- Deleting backups/tmp ... done
58 2022-11-12 17:30:46 +0000 -- Warning: Your gitlab.rb and gitlab-secrets.json files
   contain sensitive data
59 and are not included in this backup. You will need these files to restore a backup.
60 Please back them up manually.
61 2022-11-12 17:30:46 +0000 -- Backup 1668274240_2022_11_12_15.4.3 is done.
62 root@gitlab-01:/opt/gitbackup# gitlab-ctl start unicorn sidekiq
63
64 #加上配置文件
65 /var/opt/gitlab/nginx/conf #nginx配置文件
66 /etc/gitlab/gitlab.rb #gitlab配置文件
67 /etc/gitlab/gitlab-secrets.json #key文件
68
69 root@gitlab-01:/opt/gitbackup# cp /var/opt/gitlab/nginx/conf/nginx.conf ./
70 root@gitlab-01:/opt/gitbackup# cp /etc/gitlab/gitlab.rb ./
71 root@gitlab-01:/opt/gitbackup# cp /etc/gitlab/gitlab-secrets.json ./
72 root@gitlab-01:/opt/gitbackup#
73
```

### 3.4.2. 恢复

```
1 #查看备份目录
2 root@gitlab-01:/opt/gitbackup# ls -l /var/opt/gitlab/backups/
3 total 392
4 -rw----- 1 git git 399360 Nov 12 17:30 1668274240_2022_11_12_15.4.3_gitlab_backup.tar
5 root@gitlab-01:/opt/gitbackup#
```

```

1 #恢复
2 root@gitlab-01:/opt/gitbackup# gitlab-ctl stop unicorn sidekiq
3 root@gitlab-01:/opt/gitbackup# gitlab-rake gitlab:backup:restore
  BACKUP=1668274240_2022_11_12_15.4.3
4 .....
5 2022-11-12 18:05:12 +0000 -- Cleaning up
  /var/opt/gitlab/backups/terraform_state.tar.gz
6 2022-11-12 18:05:12 +0000 -- Cleaning up /var/opt/gitlab/backups/packages.tar.gz
7 2022-11-12 18:05:12 +0000 -- Deleting tar staging files ... done
8 2022-11-12 18:05:12 +0000 -- Deleting backups/tmp ...
9 2022-11-12 18:05:12 +0000 -- Deleting backups/tmp ... done
10 2022-11-12 18:05:12 +0000 -- Warning: Your gitlab.rb and gitlab-secrets.json files
  contain sensitive data
11 and are not included in this backup. You will need to restore these files manually.--+
12 2022-11-12 18:05:12 +0000 -- Restore task is done.
13 root@gitlab-01:/opt/gitbackup# gitlab-ctl start unicorn sidekiq

```

## 4. 部署jenkins服务器并安装gitlab插件、实现代码免秘钥代码clone

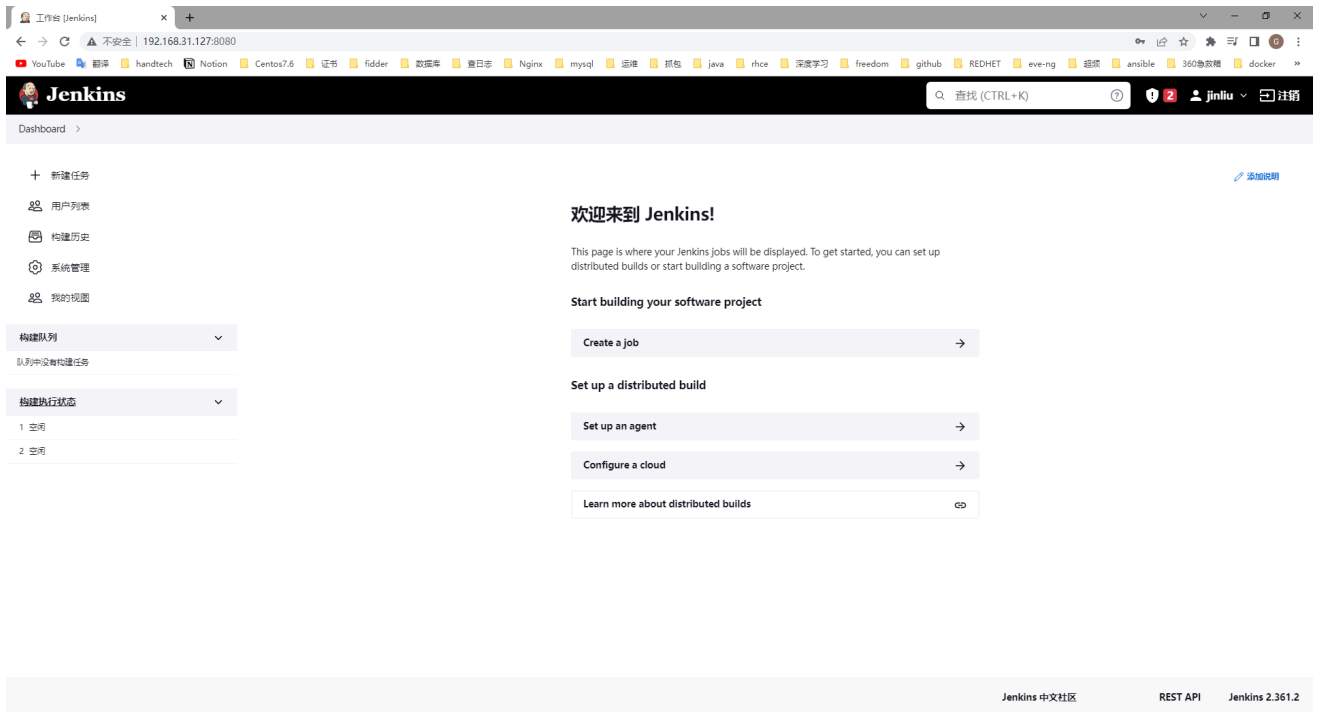
### 4.1. 安装jenkins

```

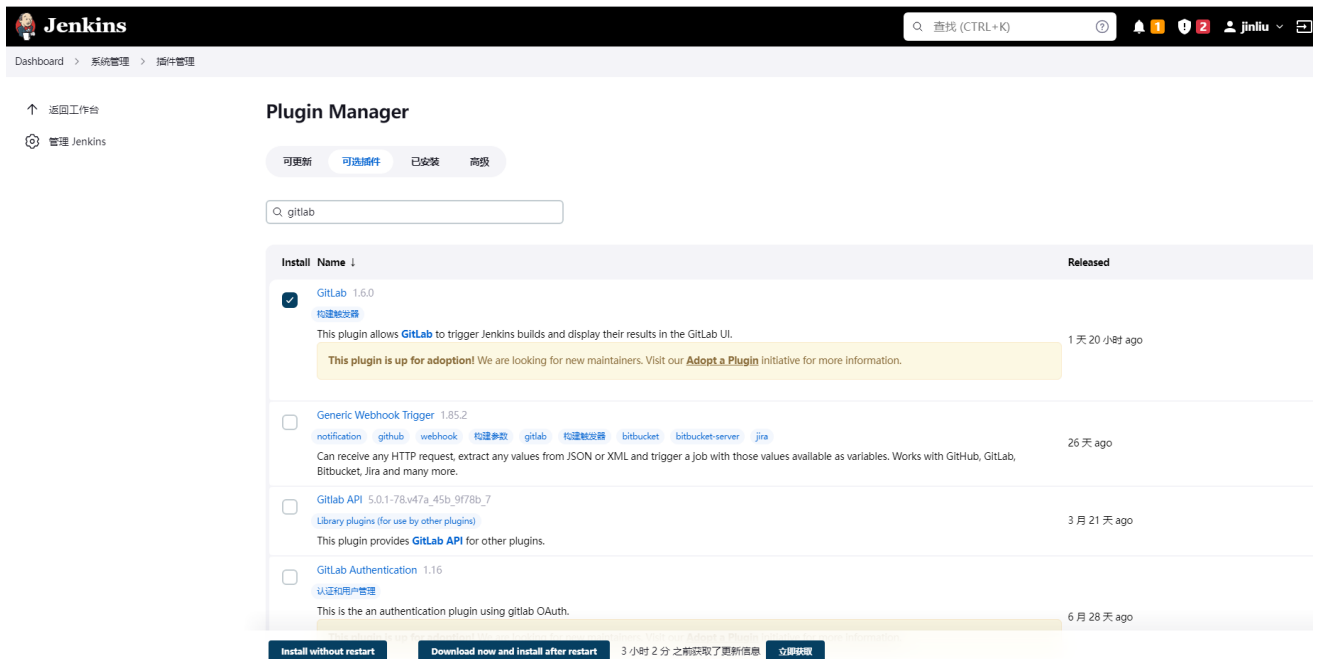
1 [root@k8s-jenkins-controller ~]# rpm -ivh jenkins-2.361.2-1.1.noarch.rpm
2 warning: jenkins-2.361.2-1.1.noarch.rpm: Header V4 RSA/SHA512 Signature, key ID
  45f2c3d5: NOKEY
3 Preparing... ##### [100%]
4 Updating / installing...
5   1:jenkins-2.361.2-1.1 ##### [100%]
6 [root@k8s-jenkins-controller ~]# yum install -y java-11-openjdk
7
8 #修改配置
9 [root@k8s-jenkins-controller ~]# vim /usr/lib/systemd/system/jenkins.service
10 # $JENKINS_WEBROOT.
11 User=root
12 Group=root
13 ...
14 # Arguments for the Jenkins JVM
15 Environment="JAVA_OPTS=-Djava.awt.headless=true -
  Dhudson.security.csrf.GlobalCrumbIssuerConfiguration.DISABLE_CSRF_PROTECTION=true"
16 -----
17 [root@k8s-jenkins-controller ~]# systemctl daemon-reload
18 [root@k8s-jenkins-controller ~]# systemctl start jenkins
19 [root@k8s-jenkins-controller ~]# ps -ef |grep jenkins
20 root      17696      1  99  17:07 ?        00:00:50 /usr/bin/java -
  Djava.awt.headless=true -
  Dhudson.security.csrf.GlobalCrumbIssuerConfiguration.DISABLE_CSRF_PROTECTION=true -
  jar /usr/share/java/jenkins.war --webroot=%C/jenkins/war --httpPort=8080
21 root      17844 16866   0  17:08 pts/0    00:00:00 grep --color=auto jenkins
22 [root@k8s-jenkins-controller ~]# systemctl enable jenkins
23 Created symlink from /etc/systemd/system/multi-user.target.wants/jenkins.service to

```

```
/usr/lib/systemd/system/jenkins.service.  
24 [root@k8s-jenkins-controller ~]#  
25 #安装默认插件并重启  
26 [root@k8s-jenkins-controller ~]# systemctl restart jenkins
```



## 4.2. 安装gitlab插件



## 4.3. 代码免秘钥代码clone

### 4.3.1. 拷贝公钥

```

1  #将jenkis上ssh公钥拷贝至gitlab服务器
2  #ubuntu机器需要更改ssh配置文件，默认无法使用root登录!!!
3  [root@k8s-jenkins-controller ~]# ssh-copy-id 192.168.31.121
4  /usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/root/.ssh/id_rsa.pub"
5  /usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter out
    any that are already installed
6  /usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompted
    now it is to install the new keys
7  root@192.168.31.121's password:
8
9  Number of key(s) added: 1
10
11  Now try logging into the machine, with:  "ssh '192.168.31.121'"
12  and check to make sure that only the key(s) you wanted were added.
13
14  [root@k8s-jenkins-controller ~]# ssh root@192.168.31.121
15  Welcome to Ubuntu 22.04.1 LTS (GNU/Linux 5.15.0-52-generic x86_64)
16
17  * Documentation:  https://help.ubuntu.com
18  * Management:    https://landscape.canonical.com
19  * Support:       https://ubuntu.com/advantage
20
21  This system has been minimized by removing packages and content that are
22  not required on a system that users do not log into.
23
24  To restore this content, you can run the 'unminimize' command.
25  Last login: Sun Nov 13 16:34:54 2022 from 192.168.31.127
26  root@gitlab-01:~# ^C
27
28  #拷贝至gitlab前端
29  [root@k8s-jenkins-controller ~]# cat .ssh/id_rsa.pub
30  ssh-rsa
    AAAAB3NzaC1yc2EAAAADAQABAAQDpCt4EQ2KfW3qrTdk2ARBN1nfFR+6Pf9s585WvtuZNoX9Rigmp7AVng
    +4cTm/VpDwo6HWlwmMSDucxHiT9HEhfHWwbdflRP6UwC2zjIqsK6k7YZET+PvNJMyrFyNgMrRayN9fNwWIKz
    dMJ5PWjlWJy0frvmrd0TgbkMoOuoPHsea2GLvtgV6W74qOsz0Lv1Xu3vMeE+5sMIAUvNrD1b91eyNez3sz7ia
    YlpxQugwbSdCcCUzSLYdvfYptnQJR9P03mNai9rcGef3oMoDvVY+uFMOMFROsKfmPjBpVlJCz7PiDMjNCbywt
    m/OPuM9RfK/zecKBvhmnDnxp9rLlE2Z7 root@k8s-jenkins-controller
31  [root@k8s-jenkins-controller ~]#
32  [root@k8s-jenkins-controller tmp]# git clone
    git@192.168.31.121:jinliudev/castillo.git
33  Cloning into 'castillo'...
34  remote: Enumerating objects: 9, done.
35  remote: Total 9 (delta 0), reused 0 (delta 0), pack-reused 9
36  Receiving objects: 100% (9/9), done.
37  Resolving deltas: 100% (1/1), done.
38  [root@k8s-jenkins-controller tmp]# ls -l
39  total 3308
40  drwxr-xr-x  3 root root      54 Nov 14 00:47 castillo
41  drwxr-xr-x. 2 root root      18 Nov 14 00:09 hsperfddata_root
42  drwx----- 2 root root       6 Nov 14 00:09 jetty-0_0_0-8080-war-_any-
    4727860105485333804
43  drwx----- 3 root root      17 Nov 13 23:26 systemd-private-

```

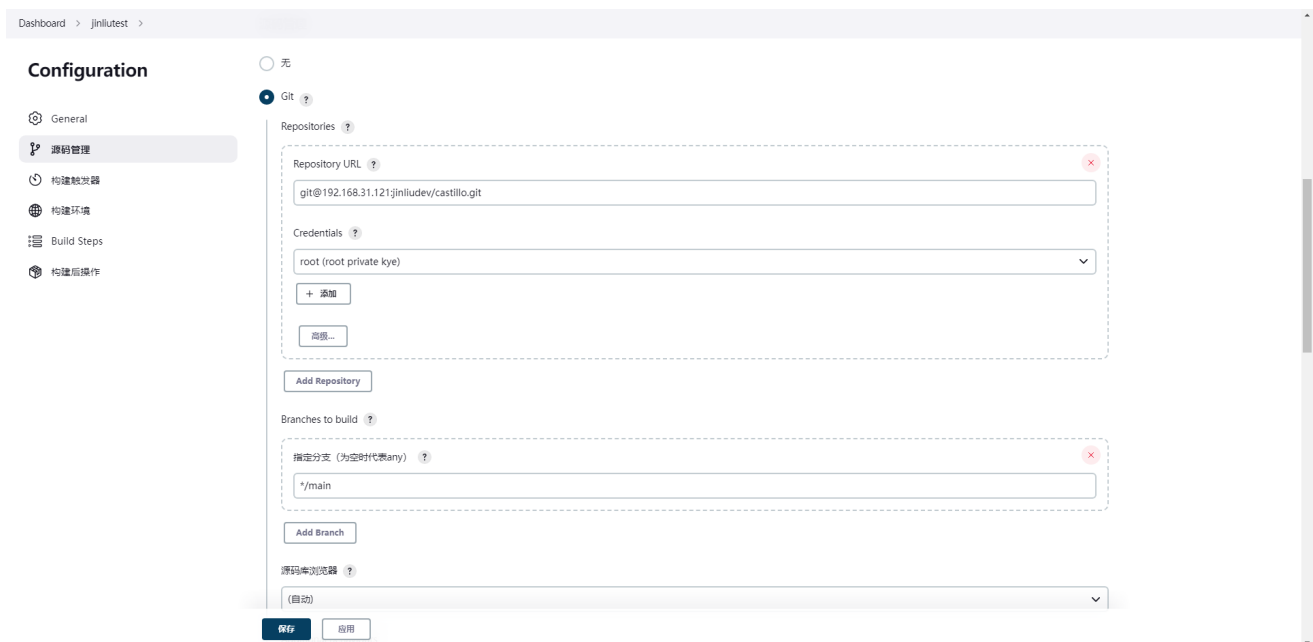
```

691936fc0dc34e64a02aed698a63edc2-chrond.service-xpHvTA
44 -rw-r--r-- 1 root root 3383324 Nov 14 00:09 winstone17659666405022763521.jar
45 [root@k8s-jenkins-controller tmp]# cd castillo/
46 [root@k8s-jenkins-controller castillo]# ls -l
47 total 12
48 -rw-r--r-- 1 root root 6203 Nov 14 00:47 README.md
49 -rw-r--r-- 1 root root 186 Nov 14 00:47 testv1.html
50 [root@k8s-jenkins-controller castillo]#

```



## 4.3.2. 配置私钥




## 4.3.3. 验证通过git拉取

```





1 [root@k8s-jenkins-controller ~]# cd /var/lib/jenkins/workspace/
2 [root@k8s-jenkins-controller workspace]# ls -l
3 total 0
4 drwxr-xr-x. 3 root root 54 Nov 14 00:57 jinliutest
5 drwxr-xr-x. 2 root root 6 Nov 14 00:57 jinliutest@tmp
6 [root@k8s-jenkins-controller workspace]# cd jinliutest
7 [root@k8s-jenkins-controller jinliutest]# ls -l

```

```
8 total 12
9 -rw-r--r-- 1 root root 6203 Nov 14 00:57 README.md
10 -rw-r--r-- 1 root root 186 Nov 14 00:57 testv1.html
11 [root@k8s-jenkins-controller jinliutest]# cat testv1.html
12 <!DOCTYPE html>
13 <html lang="en">
14 <head>
15     <meta charset="UTF-8">
16     <title>jinliu测试</title>
17 </head>
18 <body>
19     <h1>测试v1版本</h1>
20     <h1>测试v2版本</h1>
21 </body>
22 </html>
23 [root@k8s-jenkins-controller jinliutest]#
```


 **Jenkins**


查找 (CTRL+K)


   jinliu 


Dashboard > jinliutest > #6


↑ 返回到工程


 状态集


 变更记录


 **控制台输出**

 文本方式查看


 编辑编译信息

 删除构建 #6

 Git Build Data

 打开 Blue Ocean

← 上一次构建

 **控制台输出**

Started by user jinliu  
Running as SYSTEM  
Building in workspace /var/lib/jenkins/workspace/jinliutest  
The recommended git tool is: NONE  
using credential f9f8aea0-56f6-48fa-a410-de68f8a6c9af  
Cloning the remote Git repository  
Cloning repository git@192.168.31.121:jinliudev/castillo.git  
> git init /var/lib/jenkins/workspace/jinliutest # timeout=10  
Fetching upstream changes from git@192.168.31.121:jinliudev/castillo.git  
> git --version # timeout=10  
> git --version # 'git version 1.8.3.1'  
using GIT\_SSH to set credentials root private key  
Verifying host key using known hosts file  
> git fetch --tags --progress git@192.168.31.121:jinliudev/castillo.git +refs/heads/\*:refs/remotes/origin/\* # timeout=10  
> git config remote.origin.url git@192.168.31.121:jinliudev/castillo.git # timeout=10  
> git config --add remote.origin.fetch +refs/heads/\*:refs/remotes/origin/\* # timeout=10  
Avoid second fetch  
> git rev-parse refs/remotes/origin/main^{commit} # timeout=10  
Checking out Revision 3601a17d5066e560d58cee853663eddc2aabb6c7 (refs/remotes/origin/main)  
> git config core.sparsecheckout # timeout=10  
> git checkout -f 3601a17d5066e560d58cee853663eddc2aabb6c7 # timeout=10  
Commit message: "v2"  
First time build. Skipping changelog.  
[jinliutest] \$ /bin/sh -xe /tmp/jenkins13512955004654866930.sh  
Finished: SUCCESS

Jenkins 中文社区

REST API

Jenkins 2.361.2