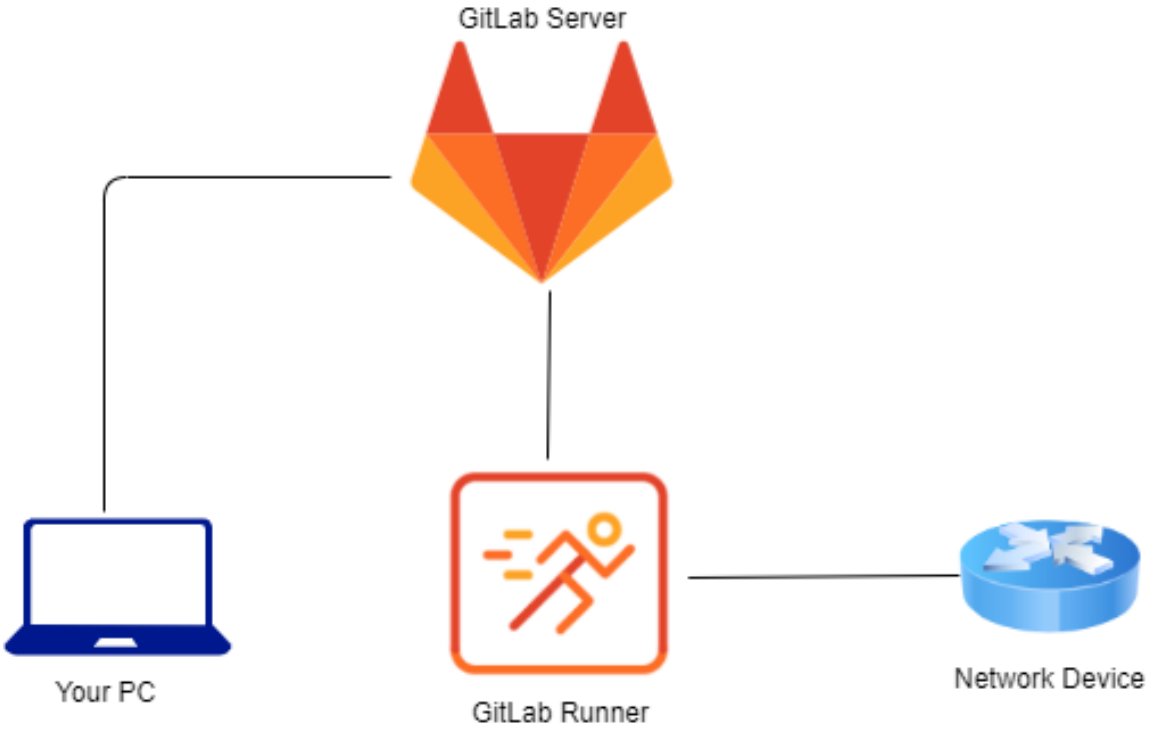
# Lab - Sử dụng Gitlab CI cấu hình hostname cho thiết bị

## 1. Mô hình LAB




Hình 1: Mô hình LAB sử dụng để thực thi.

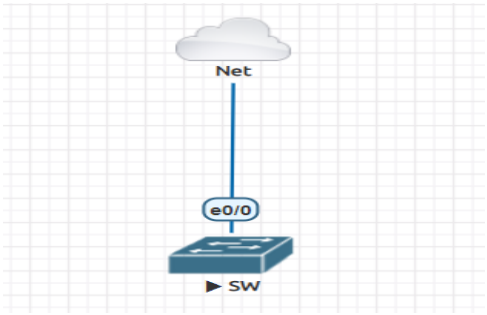
## 2. Yêu cầu

Thực hiện xây dựng luồng CI/CD cơ bản để thực hiện deploy thành công cấu hình hostname cho thiết bị với gitlab server và Gitlab Runner.

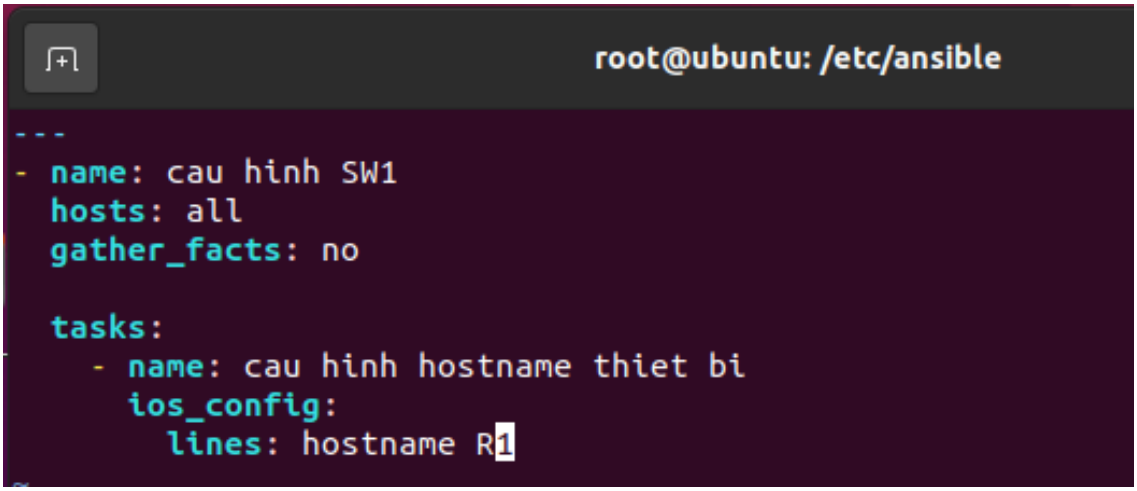
## 3. Các bước thực hiện

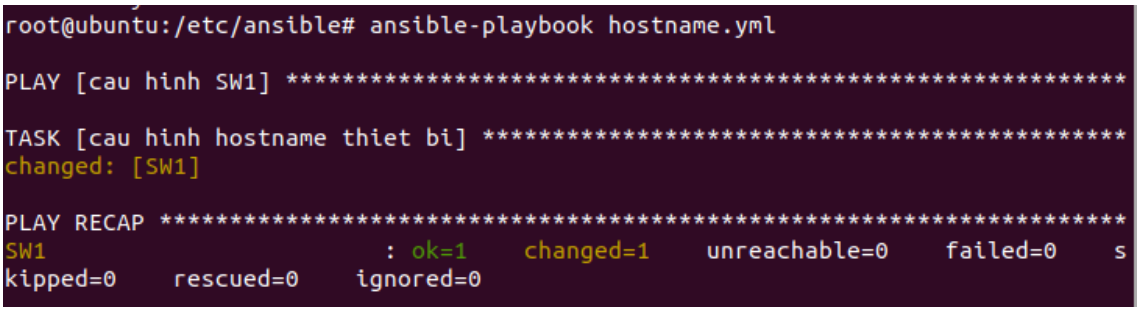
- Sau khi thực hiện cài đặt thành công Gitlab Server và Gitlab Runner ở bài lab trước chúng ta tiếp tục sử dụng chúng để cấu hình hostname cho thiết bị.

- Thiết bị cần có kết nối SSH để thực hiện cấu hình:



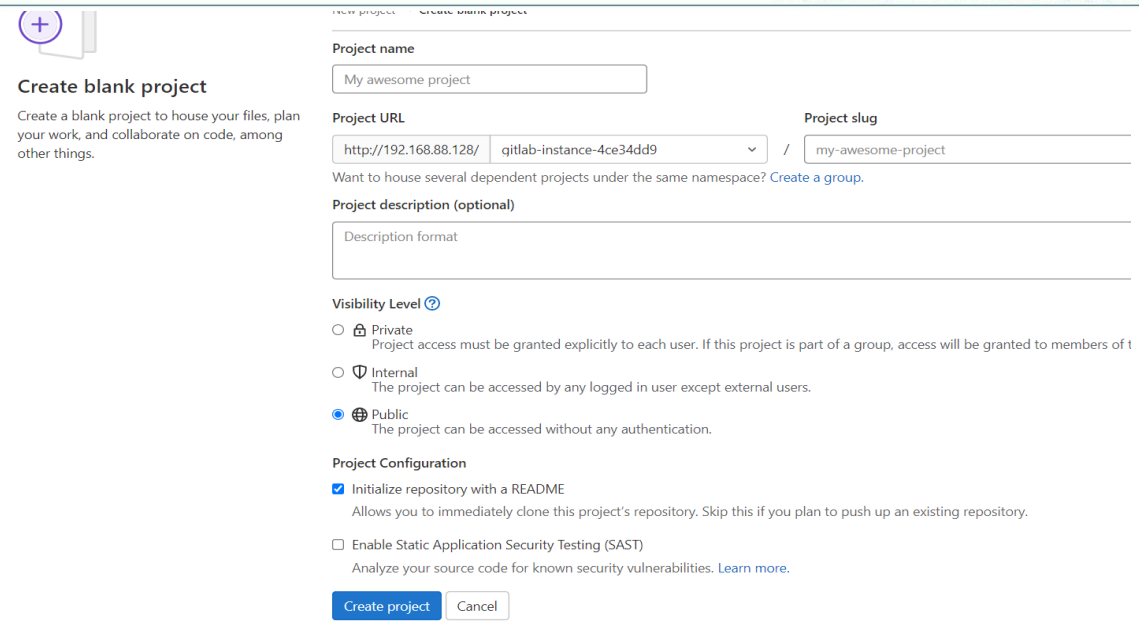
- Tiếp theo thực hiện sử dụng Ansible cấu hình hostname cho thiết bị như ở lab 31 và thực hiện chạy thành công.



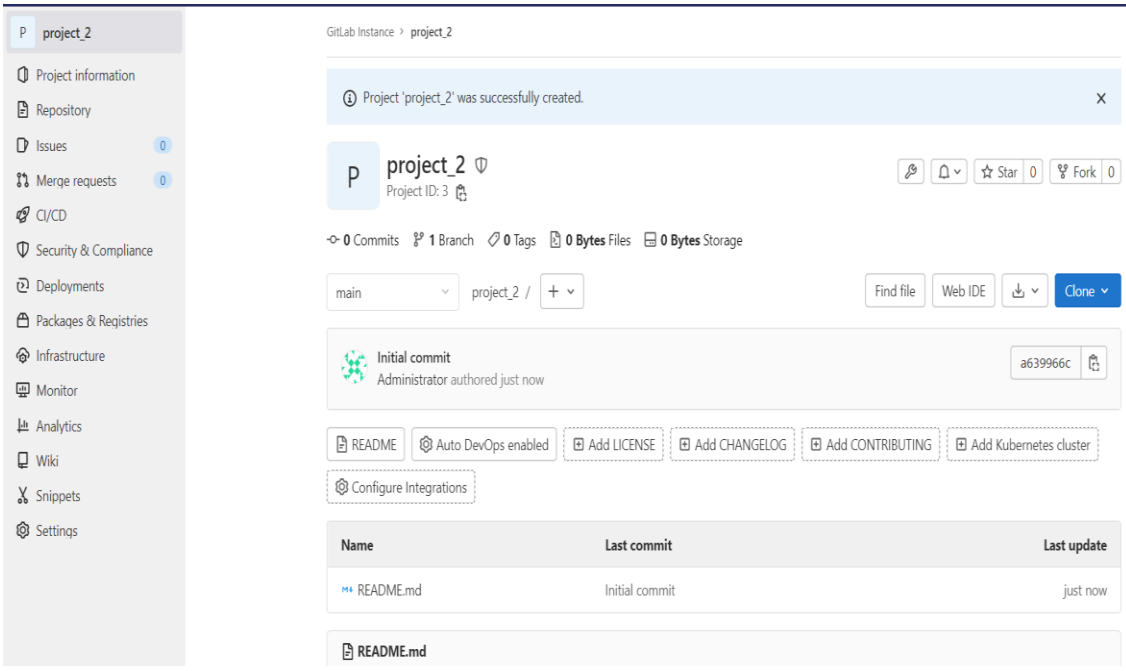




- Thực hiện tạo project trên Gitlab Server và thực hiện kết nối Gitlab Runner với Server:

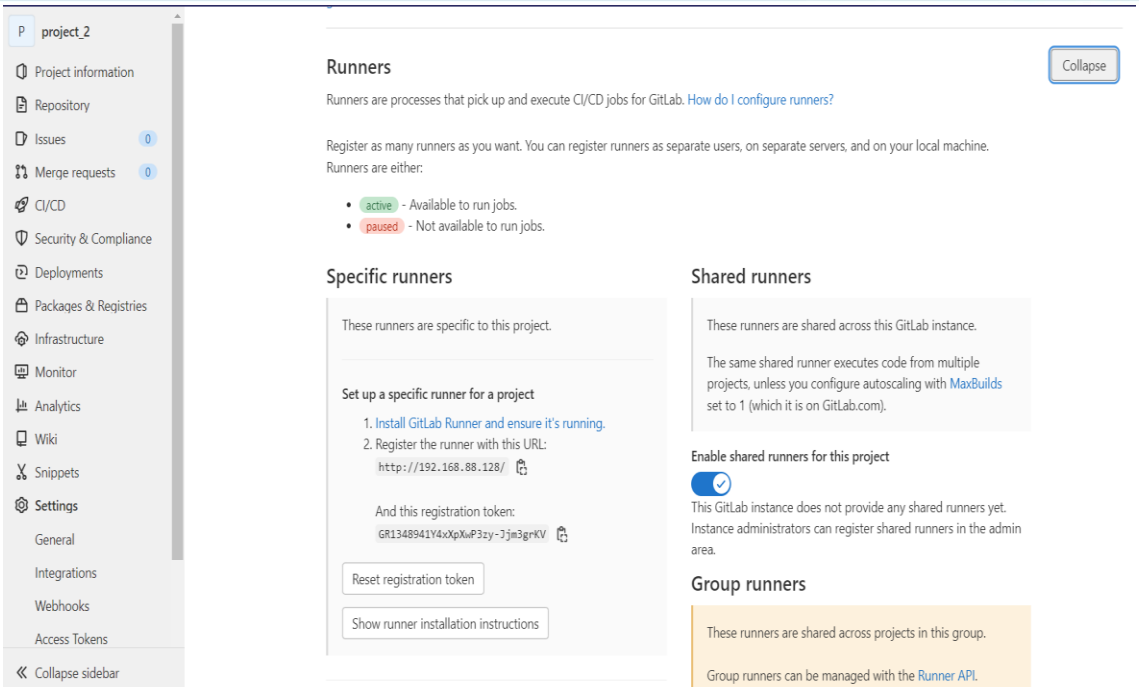
+ Tạo project như đã tạo đối với Lab: CI/CD cơ bản




+ Tạo project thành công:




+ Thực hiện vào Settings → CI/CD → Runner

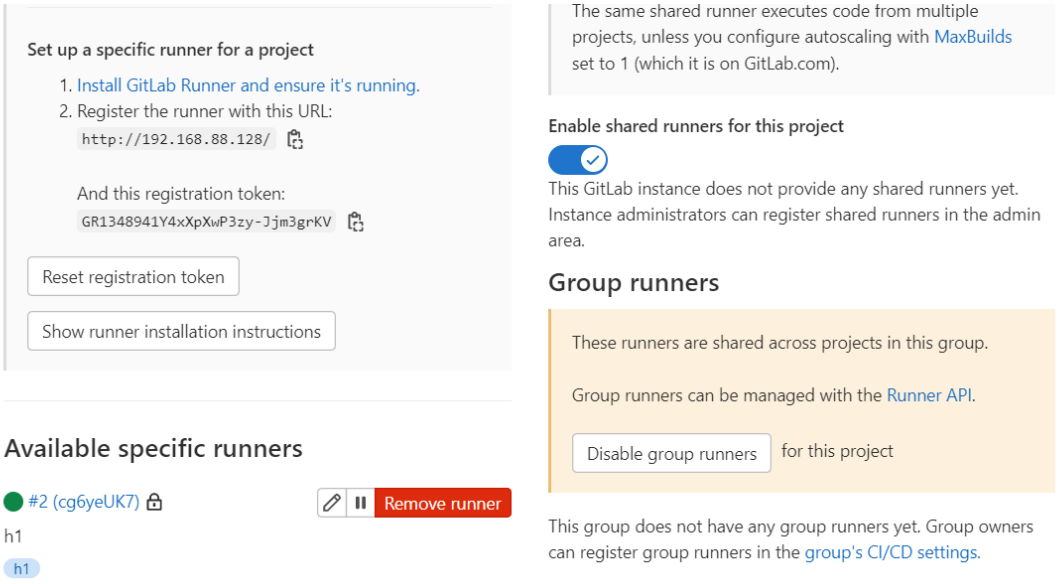



- Thực hiện kết nối Runner với Server ở máy ubuntu:

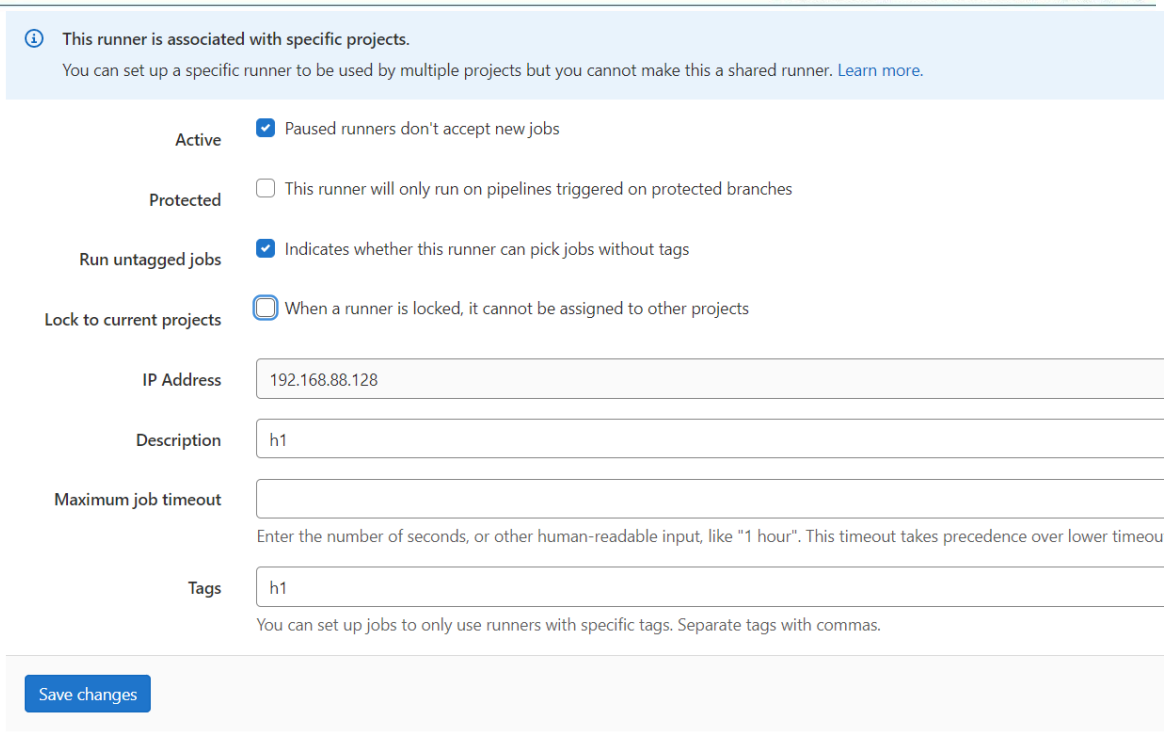
```
root@ubuntu:/home/huuhiep/Desktop/project_2# gitlab-runner register
Runtime platform                                    arch=amd64 os=linux pid=3515
 revision=76984217 version=15.1.0
Running in system-mode.

Enter the GitLab instance URL (for example, https://gitlab.com/):
http://192.168.88.128/
Enter the registration token:
GR1348941Y4xXpXwP3zy-Jjm3grKV
Enter a description for the runner:
[ubuntu]: h1
Enter tags for the runner (comma-separated):
h1
Enter optional maintenance note for the runner:
h1
Registering runner... succeeded                     runner=GR1348941Y4xXpXwP
Enter an executor: custom, ssh, virtualbox, docker+machine, docker-ssh+machine,
docker, docker-ssh, parallels, shell, kubernetes:
docker
Enter the default Docker image (for example, ruby:2.7):
centos:centos7.9.2009
Runner registered successfully. Feel free to start it, but if it's running alrea
dy the config should be automatically reloaded!
```

( Ghi chú: nhập các url và token để kết nối cũng như điền tên cần thiết để thực hiện viết file và chọn docker làm nơi thực thi trên đó)

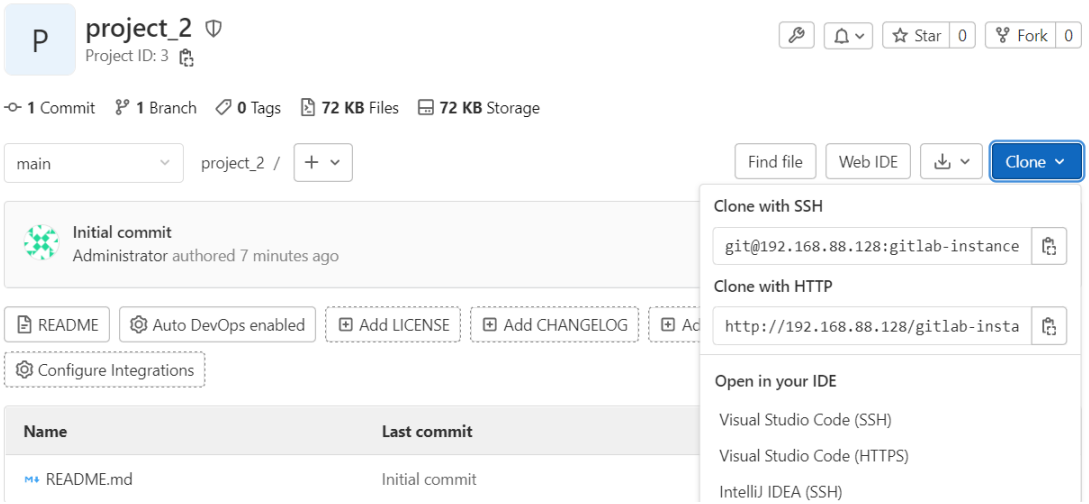
- Kiểm tra Runner đã hoạt động hay chưa bằng cách reload lại trang:




- Và thực hiện chỉnh sửa ở hình chỉnh sửa:

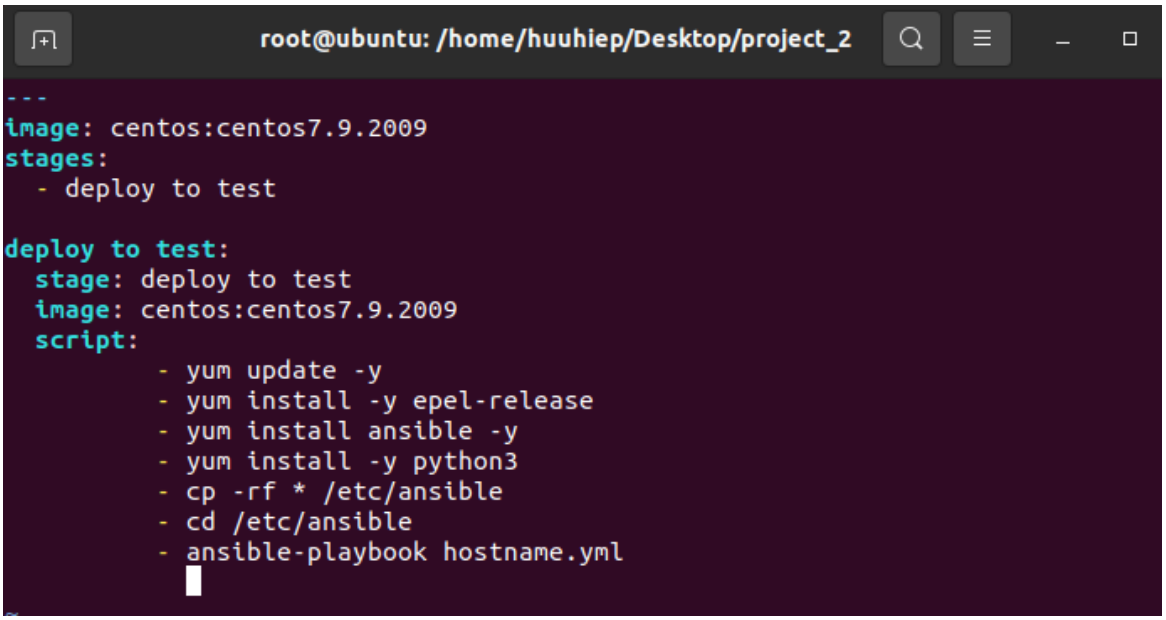



- Tiếp theo ta thực hiện clone project đã tạo về máy và truy cập vào thư mục:

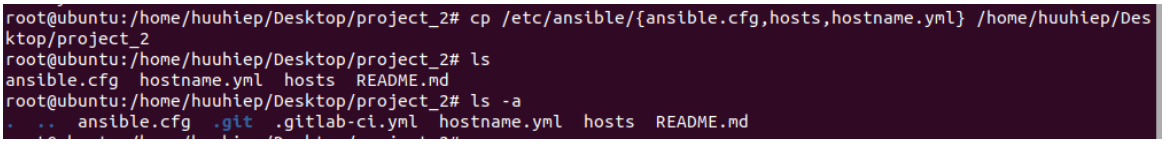



```
root@ubuntu:/home/huuhiep/Desktop# git clone http://192.168.88.128/gitlab-instan
ce-4ce34dd9/project_2.git
Cloning into 'project_2'...
Username for 'http://192.168.88.128': root
Password for 'http://root@192.168.88.128':
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Compressing objects: 100% (2/2), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (3/3), 2.79 KiB | 2.79 MiB/s, done.
root@ubuntu:/home/huuhiep/Desktop# cd project_2
root@ubuntu:/home/huuhiep/Desktop/project_2#
```

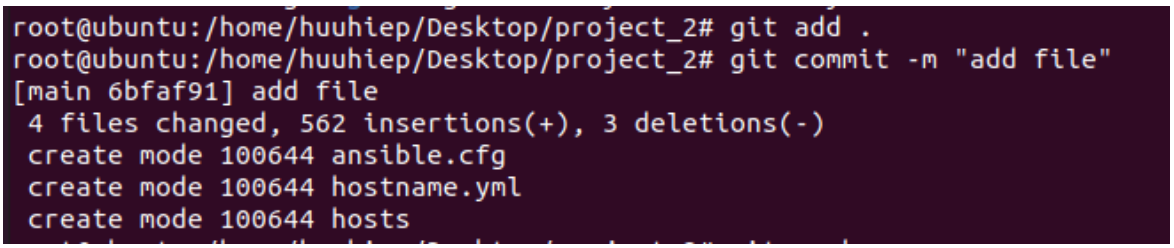
- Thực hiện tạo file .gitlab-ci.yml để thực thi cấu hình hostname:

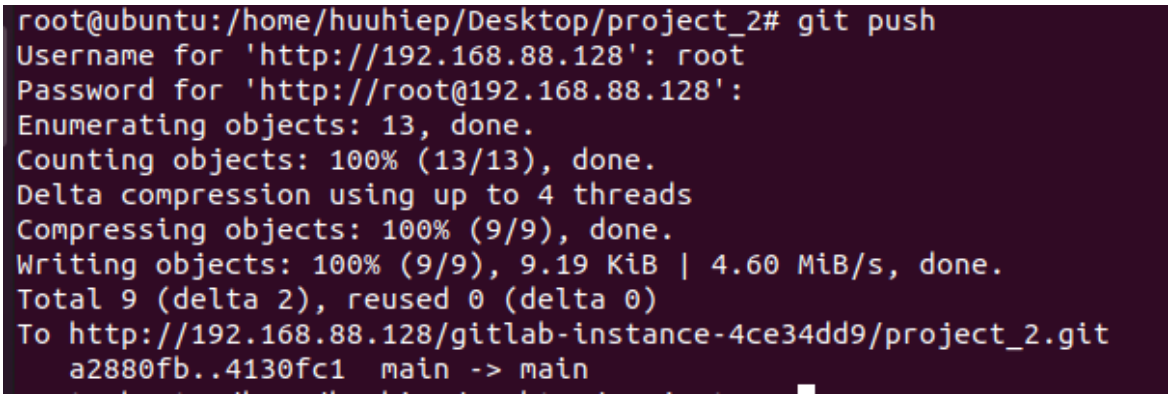



- Thực hiện copy các file ansible.cfg, hosts, hostname.yml vào project đã clone về:

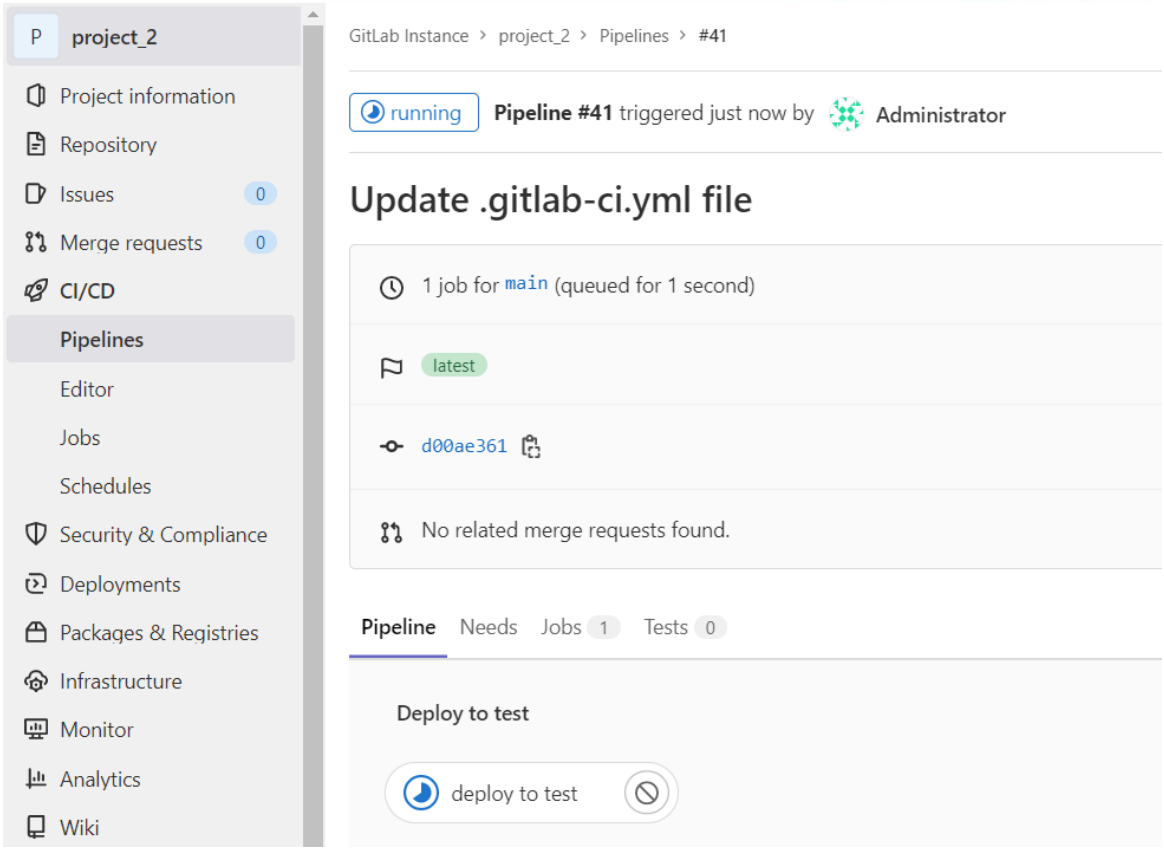



- Sau đó thực hiện push code lên:




- Kiểm tra kết quả thông qua pipeline:




```
716   python3-pip.noarch 0:9.0.3-8.el7   python3-setuptools.noarch 0:39.2.0-10.el7
717 Complete!
718 $ cp -rf * /etc/ansible
719 $ ansible-playbook hostname.yml
720  [WARNING] Ansible is being run in a world writable directory (/builds/gitlab-instance-4ce34dd9/project_2), ignoring it as
    an ansible.cfg source. For more information see https://docs.ansible.com/ansible/devel/reference_appendices/config.html#cfg-
    in-world-writable-dir
721 PLAY [cau hinh SW1] ***********************************************************
722 TASK [cau hinh hostname thiet bi] *********************************************
723 ok: [SW1]
724 PLAY RECAP *********************************************************************
725 SW1                        : ok=1    changed=0    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
726 Job succeeded
```

→ Kiểm tra trên thiết bị thành công.


R1>
R1>
R1>
R1>
R1>
R1>
R1>
R1>
R1>