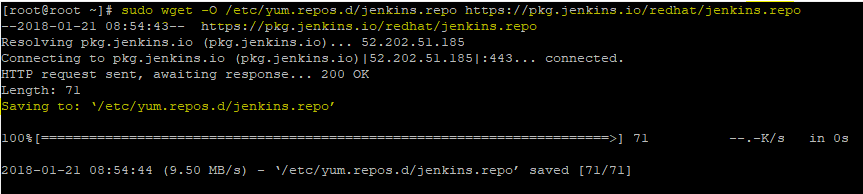
**CÀI ĐẶT VÀ CẤU HÌNH JENKINS SERVER**

**1. CÀI ĐẶT JENKINS**

**1.1 Tạo repo Jenkins & import GPL Key cho YUM**



sudo wget -O /etc/yum.repos.d/jenkins.repo https://pkg.jenkins.io/redhat-stable/jenkins.repo

Lệnh Wget với option -O sẽ giúp bạn tải file jenkins.repo và paste file đó vào đường dẫn chỉ định là /etc/yum.repo.d/jenkins.repo

rpm --import <https://pkg.jenkins.io/redhat-stable/jenkins.io.key>



**1.2 Cài đặt môi trường JAVA cho Jenkins**

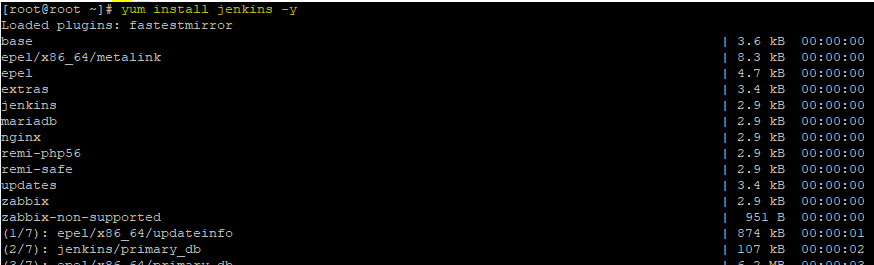
yum install java -y



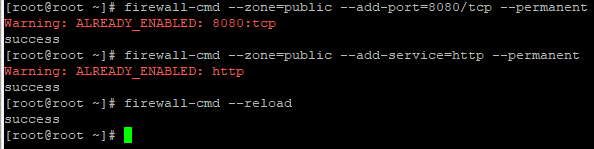
Sau khi cài đặt xong, để xác thực việc môi trường Java đã cài đặt thành công, ta kiểm tra phiên bản Java bằng câu lệnh: java -version.

**1.3 Tiến hành cài đặt Jenkins bằng Yum**

yum install jenkins -y



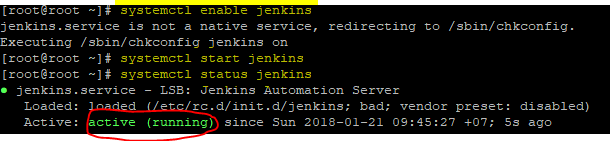
**1.4 Vô hiệu hóa Firewall cho phép Jenkins hoạt động.**



sudo firewall-cmd --zone=public --permanent --add-port=8080/tcp

sudo firewall-cmd --reload

**1.5 Bật service Jenkins và start và kiểm tra trạng thái service Jenkins.**

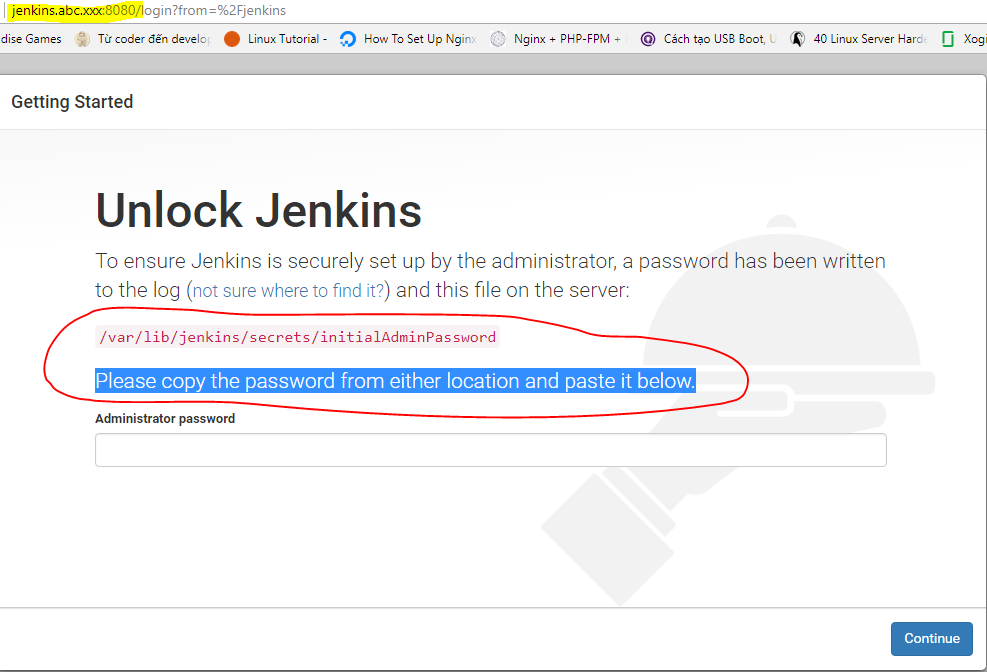


Hiển thị trạng thái active (running) có nghĩa là Jenkins bạn đã sẵn sàng để sử dụng.

**2. Sử dụng Jenkins.**

**2.1 Truy cập vào giao diện http của Jenkins**

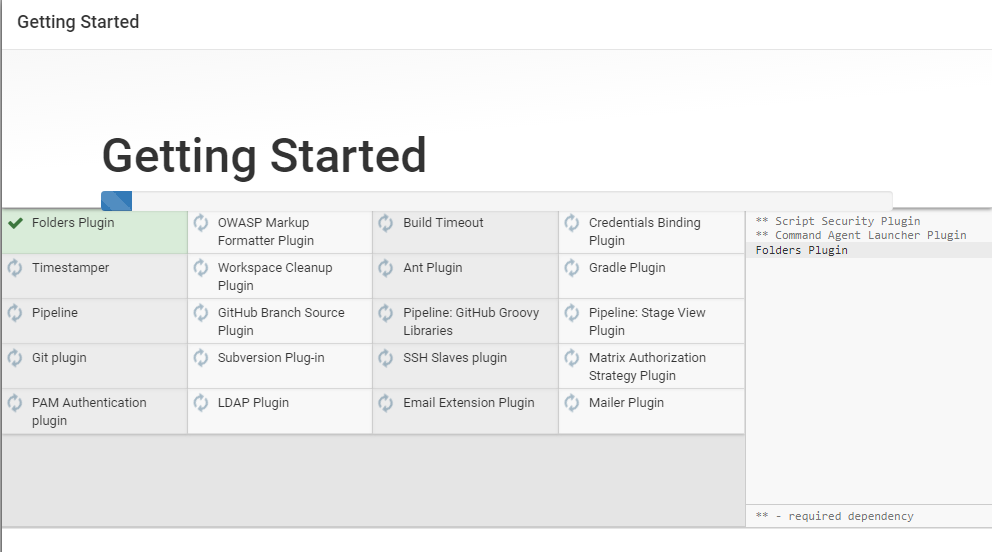
Sử dụng: localhost:8080 hoặc nếu máy chủ của bạn có tên miền thì truy cập bằng tenmiencua.ban:8080



Copy chuỗi password và paste vào giao diện administrator password 🡪 Continue



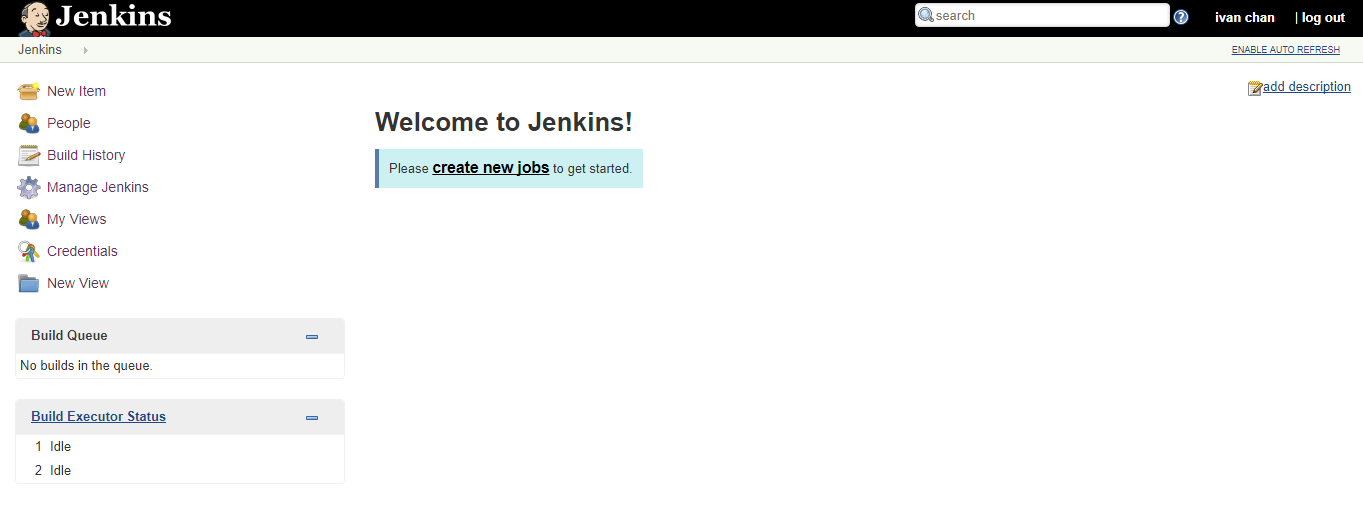
Bước đầu yêu cầu cài đặt các Plugin cho Jenkins. Chọn Suggest Plugin for Jenkins



Tiếp theo là tạo user quản trị đầu tiên.

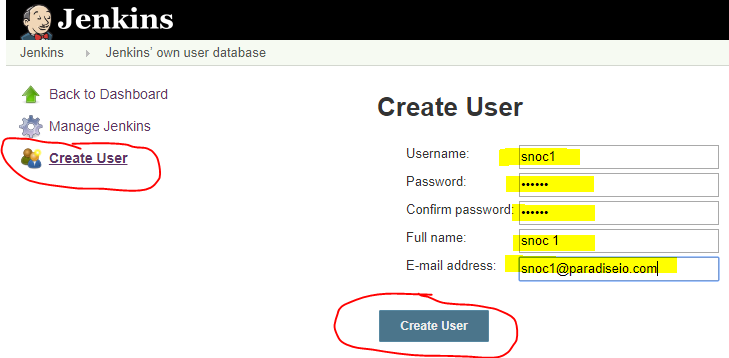


Đăng nhập vào user vừa tạo, đây là giao diện sơ khai của Jenkins.



**2.2 Tạo 1 User mới trong Jenkins.**

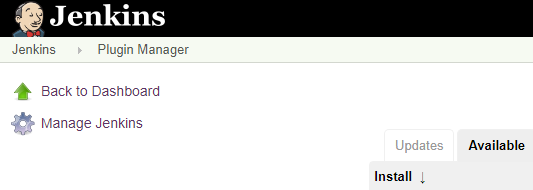
**Để tạo user mới, ta vào Manage Jenkins**  **🡪 Manage User** 



**2.3 Tạo theme (giao diện) theo ý muốn**

**Truy cập vào trang *https://plugins.jenkins.io/simple-theme-plugin***

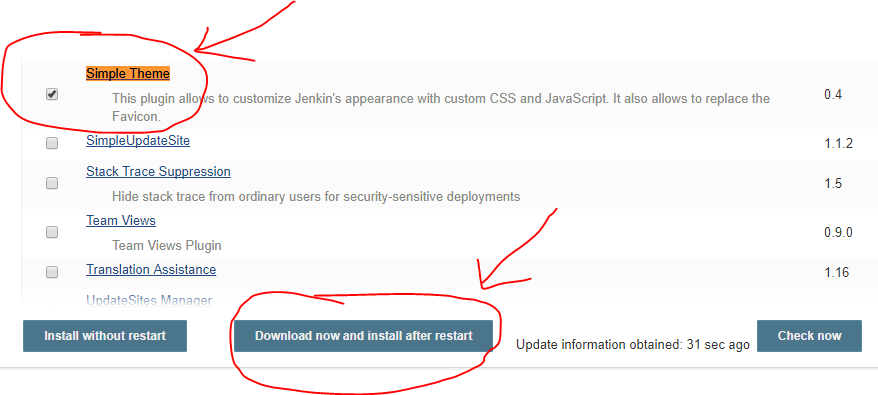
**B1: Tải plugin về và cài đặt trong Jenkins server (trong mục Plugin Manager). Truy cập vào trang Tìm hiểu về plugin sẽ sử dụng để cài đặt theme. *https://plugins.jenkins.io/simple-theme-plugin***



**Search “Simple Theme” plugin và tiến hành cài đặt, sau đó restart service Jenkins bằng trình duyệt:**

[**http://ivan-jenkins.paradiseio.com:8080/saferestart**](http://ivan-jenkins.paradiseio.com:8080/saferestart) **(đợi chạy hết job rồi restart jenkins)**

[**http://ivan-jenkins.paradiseio.com:8080/restart**](http://ivan-jenkins.paradiseio.com:8080/restart) **(Restart jenkins tức thời)**

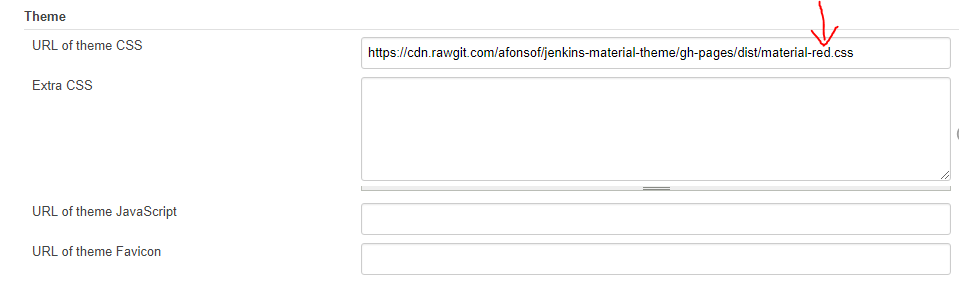


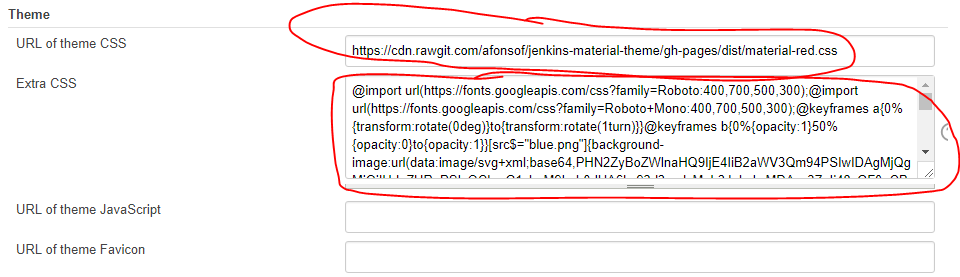
**B2: Tạo file .CSS theo ý muốn tự chọn.**

**- Vào trang** [**http://afonsof.com/jenkins-material-theme/**](http://afonsof.com/jenkins-material-theme/)

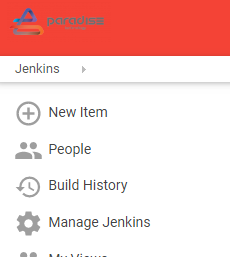
**- Chọn màu sắc và gõ tên màu vào link giống hình dưới:**



 **- Logo công ty (upload từ máy tính lên trang web tạo theme trên) 🡪 download file .CSS về máy, copy nội dung file CSS đưa vào mục extra CSS giống trong hình dưới:**

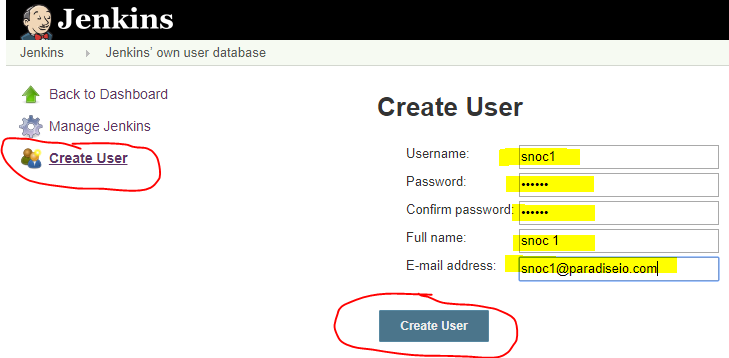


**- Save lại config và truy cập trang chủ Jenkins kiểm chứng (đã có màu mới và logo mới):**



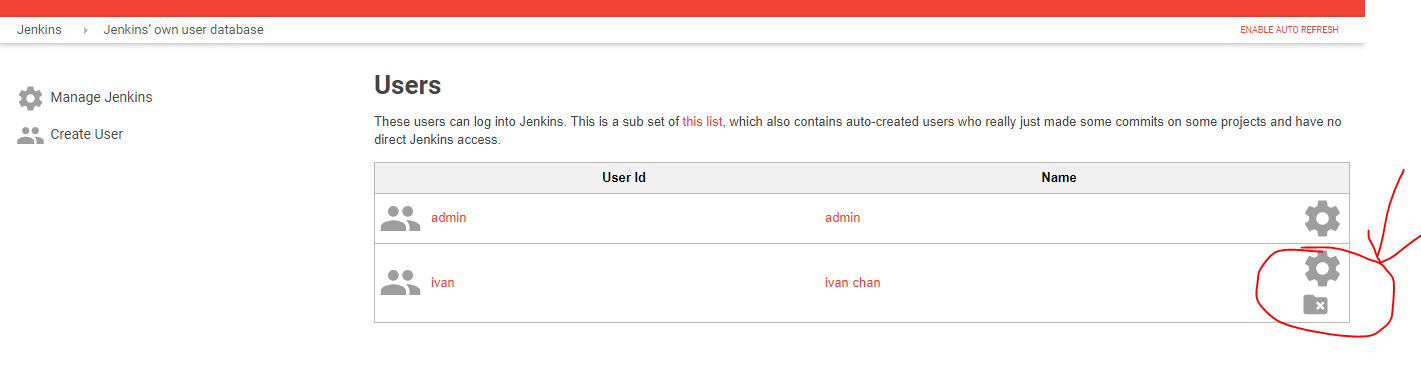
**2.4 Quản lý User**

**- Tạo User: Để tạo user mới, ta vào Manage Jenkins**  **🡪 Manage User** 



**- Chỉnh sửa User:**

**Vào Jenkins config 🡪 Manage User 🡪 click vào nút edit theo hình để chỉnh sửa profile cho user**



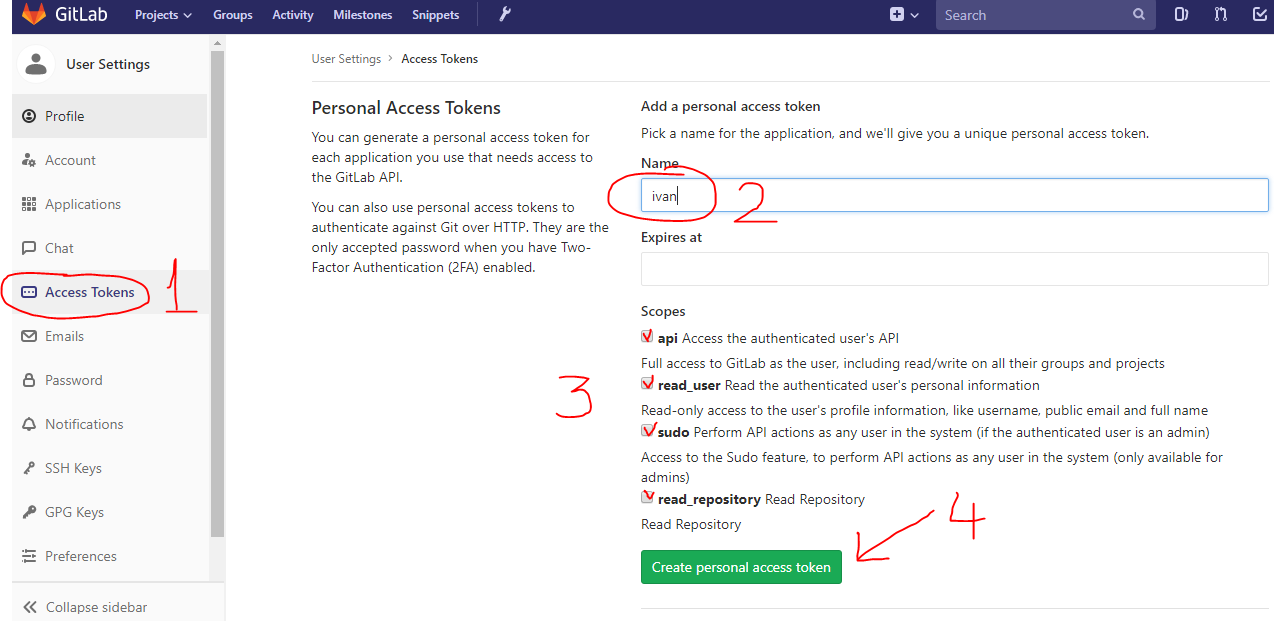
**2.4 Tạo một Job mới trong Jenkins**

**2.4.1 Tạo Job Jenkins tích hợp Gitlab (git source)**

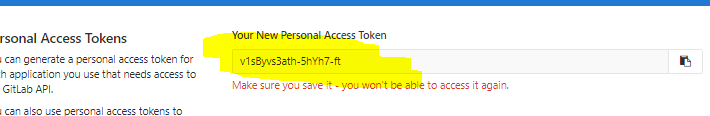
**- Cài plugin “Gitlab plugin” cho Jenkins**

**Vào phần Plugin manager để tìm và cài đặt, sau đó restart jenkins service (cách cài plugin giống mục 2.3)**

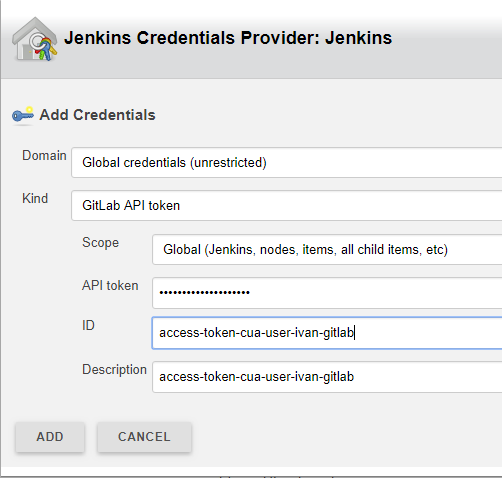
**- Qua gitlab tạo 1 user: ivan, sau đó tạo Access token cho user IVAN theo hình hướng dẫn sau:**



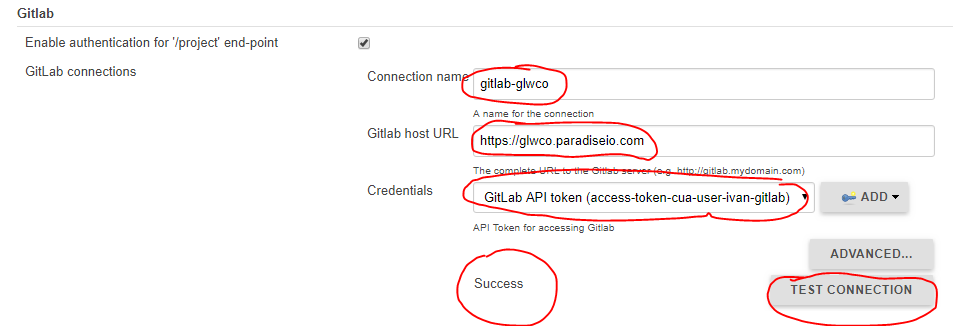
**Sau đó copy Access Token**



**Qua máy Jenkins, tạo Credentials dạng Gitlab API token**



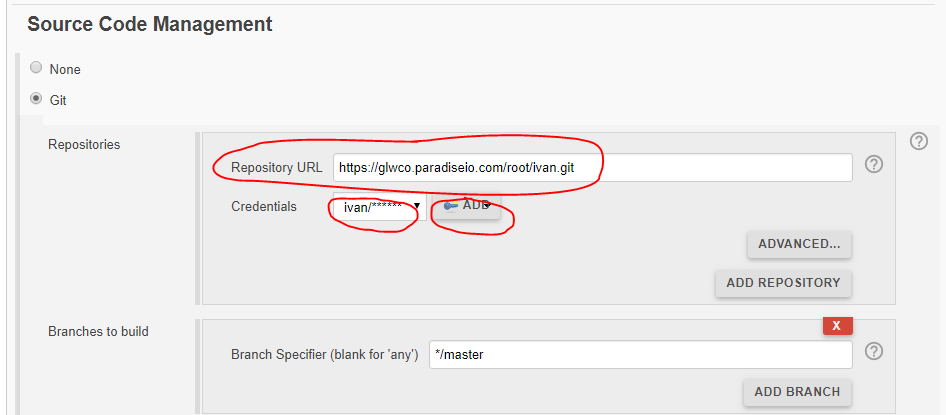
**Sau đó vào phần Jenkins configure, kéo xuống khu vực gitlab, nhập vào giống hình:**



**Như vậy là JENKIN tích hợp GITLAB đã liên kết thành công.**

**- Tạo job gitlab (tự động clone và pull repo trên gitlab về workspace máy jenkins)**

**Làm theo thứ tự sau: tạo 1 job freestyle, đến mục Source Code Management thì chọn Git**

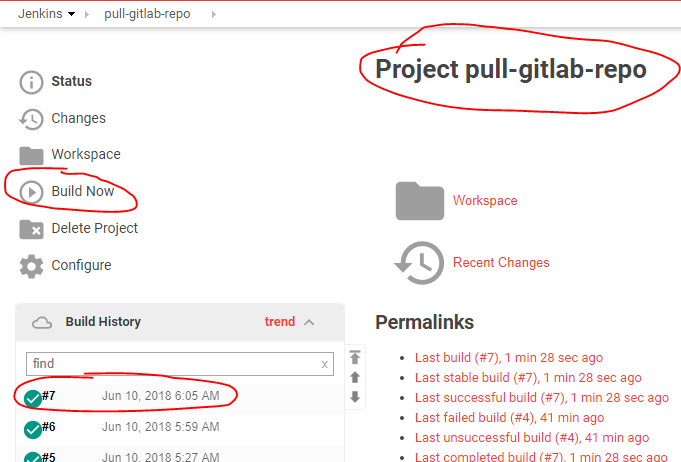


**Repo: Đường dẫn project trên gitlab**

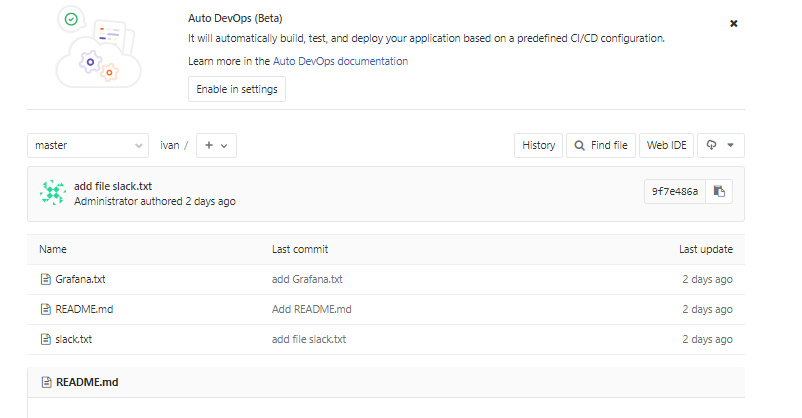
**Credentials: Tạo 1 creadentials dạng user/pass của user IVAN (đã tạo trên máy gitlab)**

**Cuối cùng save lại Job.**

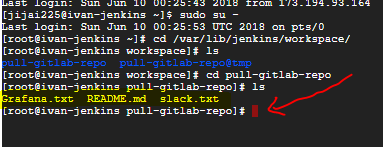
**Như vậy, khi có user gitlab nào commit thay đổi trên gitlab, bạn có thể dùng Jenkin chạy job để pull về workspace.**



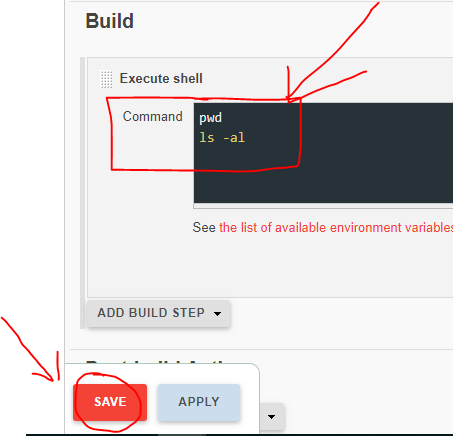
**Kiểm tra Repo trên máy gitlab:**



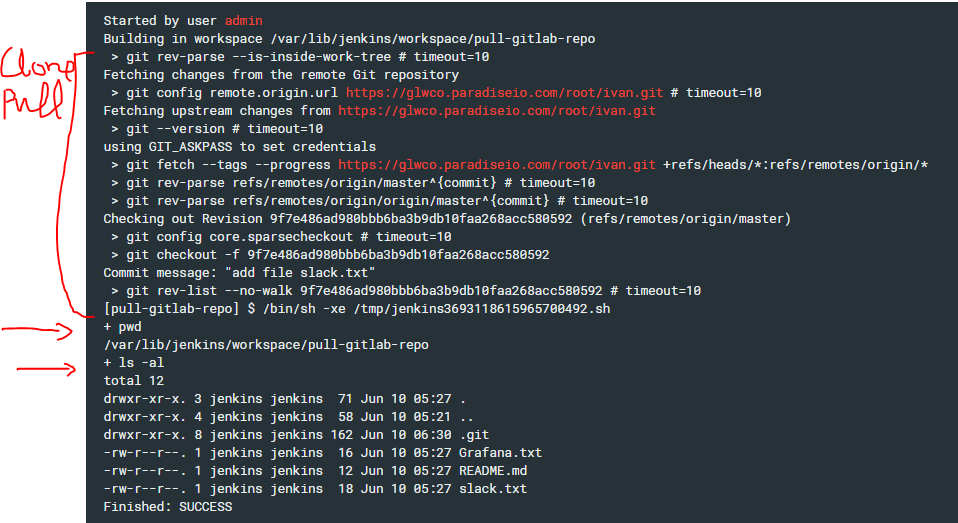
**Kiểm tra Repo đã clone và pull về vào máy Jenkin ở workspace, như vậy ta đã có repo mang vào xử dụng.**



**Sau khi clone và pull về, ta tiếp tục chạy build “Execute Shell” như hình sau:**



**Xem kết quả của job tại console output:**



**Bài tập: Áp dụng kiến thức này để chạy script check\_SSL sau khi commit file domain và file sh từ gitlab. (Tiếp tục thực hành bài tập dưới dựa trên 2 máy Jenkins & Gitlab đã dựng ở phần trên)**

**Hướng dẫn:**

**A Trên máy gitlab, tiến hành tạo 1 repo mới, tạo 2 file & commit 2 files**

1. file “domain”

Nội dung:

Google.com

2. file “[check\_ssl.sh](http://gitlab.test.com/root/check_ssl_repo/blob/master/check_ssl.sh)”

Nội dung:

#!/bin/bash

echo -e "Domain|Valid Date|Expired Date|Remaining Days|Issuer" > ./temp

for domain in $(cat ./$1)

do

echo "Dang kiem tra SSL domain $domain..."

echo | openssl s\_client -servername $domain -connect $domain:443 2>/dev/null | openssl x509 -noout 2>/dev/null

if [ $? -eq 0 ];then

start=$(echo | openssl s\_client -servername $domain -connect $domain:443 2>/dev/null | openssl x509 -noout -dates | grep notBefore | cut -d'=' -f2)

end=$(echo | openssl s\_client -servername $domain -connect $domain:443 2>/dev/null | openssl x509 -noout -dates | grep notAfter | cut -d'=' -f2)

d=$(date -d "$end" +%s)

now=$(date +%s)

remain=$(echo $(( (d - now) / 86400 ))days)

issue=$(echo | openssl s\_client -servername $domain -connect $domain:443 2>/dev/null | openssl x509 -noout -issuer | grep issuer )

echo -e "$domain|$start|$end|$remain|$issue" >> ./temp

else

echo -e "$domain|invalid|invalid|invalid|invalid" >> ./temp

fi

done

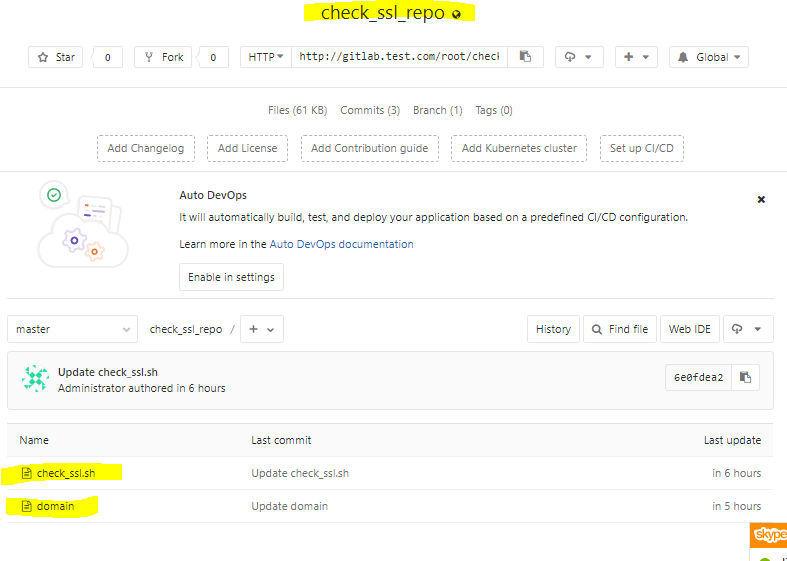
column -s "|" -o "," -t ./temp > $1\_ssl\_result.csv

column -s "|" -t ./temp > $1\_ssl\_result

rm -rf ./temp

echo "Da hoan thanh kiem tra SSL toan bo file $1"

Sau đó commit 2 file này lên Repo. (xem hình dưới)



**B. Máy Jenkins tạo 1 freestyle job.**

1. Phần source code management:

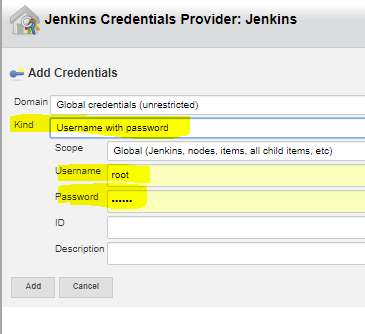
Credentials: tạo 1 credential là cặp user/pass của user bên gitlab lấy qua.

Click vào add, Kind: Username with password

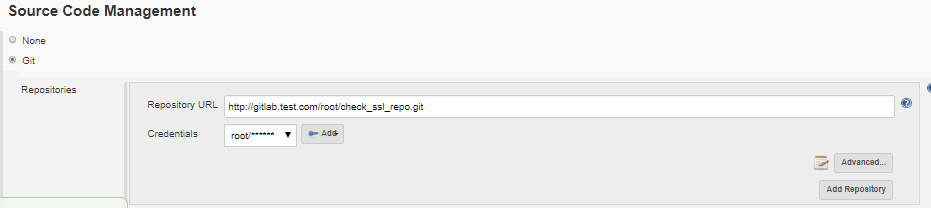
Username: (user của gitlab)

Password: (password của user gitlab bên trên)

Chọn Add



Sau đó trong phần Credentials chọn cái Credentials vừa tạo bước trên.



2. Phần build:

Thêm 2 dòng commandline như hình dưới

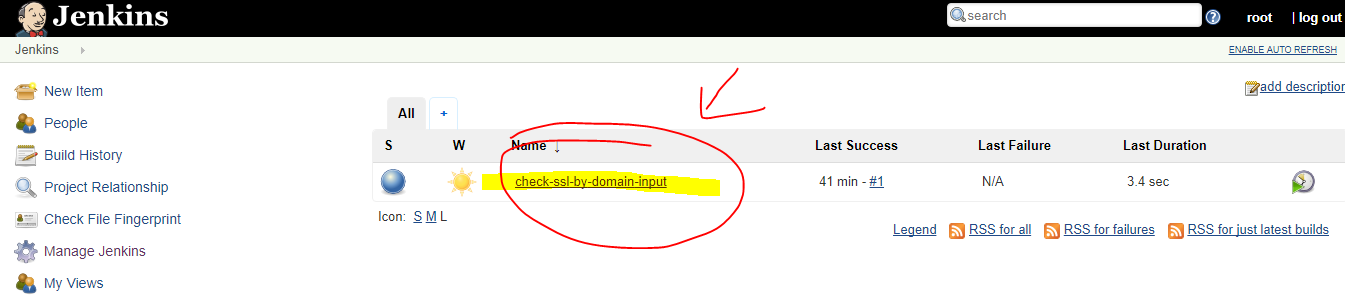


3. Phần general:

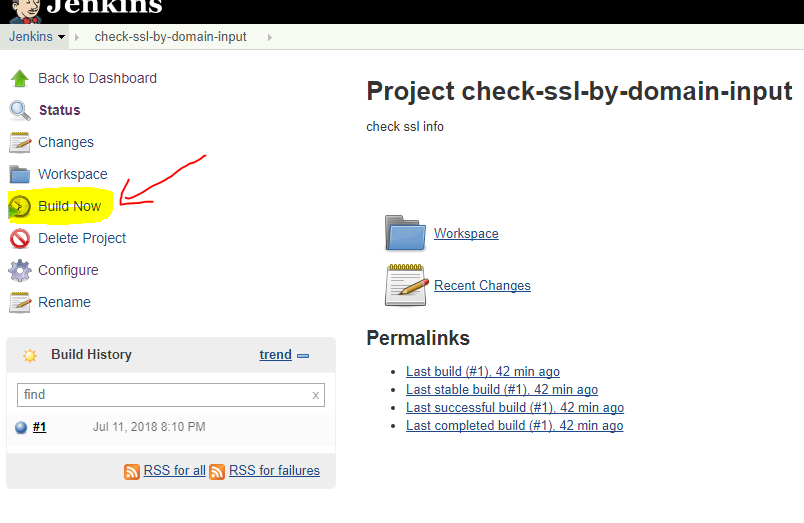
Không quan trọng, chỉ cần điền chú thích vào dòng descriptions.

**C. Chạy build job trên máy Jenkins**

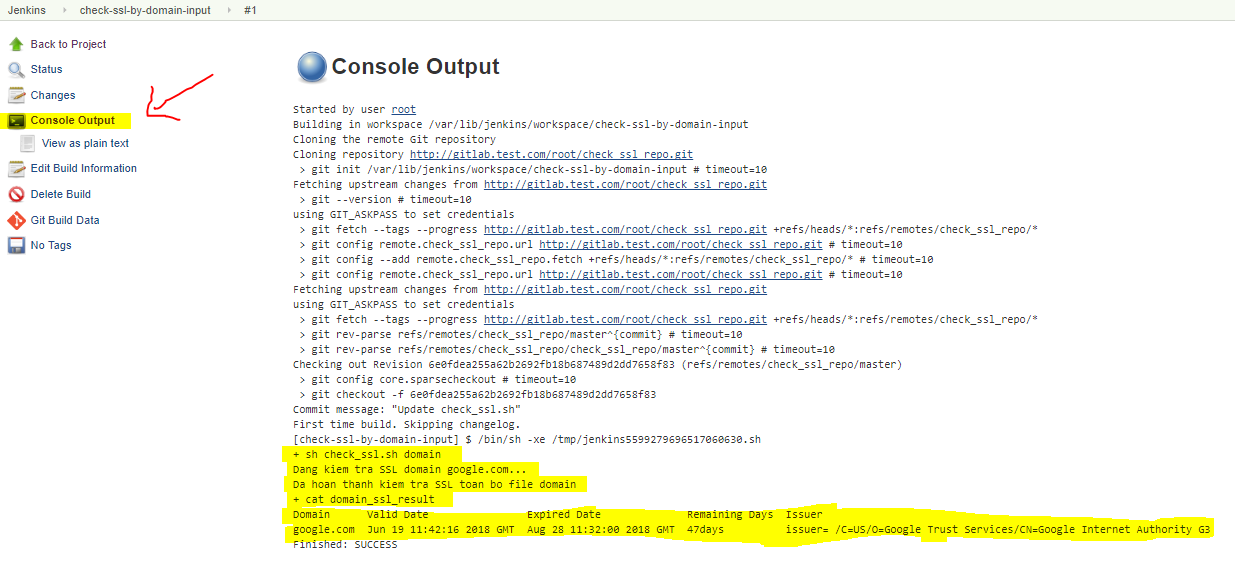
Click vào jenkins job



Sau đó nhấn nút Build Now



Sau đó click vào console output để kiểm tra kết quả check ssl cho file domain.



Jenkins đã chạy job thành công và trả về kết quả check SSL trong console output.

1 So link tham khao

1/ <https://viblo.asia/p/gioi-thieu-ve-jenkins-mot-open-source-continuous-integration-server-zb7vDV0OMjKd>

2/ <https://kipalog.com/posts/Su-dung-Jenkins-de-deploy-code-co-su-approval-cua-manager>

3/ <https://cuongquach.com/jenkins-huong-dan-reset-mat-khau-user-jenkins-admin.html>

4/ <https://blog.vietnamlab.vn/2017/11/22/pipeline-voi-jenkins/>