

## Lab - Setup CI trên Jenkins

### Thực hiện cài đặt Jenkins trên máy ảo

- Đối với máy ảo Centos:

<https://www.liquidweb.com/kb/how-to-install-jenkins-on-centos-8/>

- Đối với Ubuntu:

<https://www.digitalocean.com/community/tutorials/how-to-install-jenkins-on-ubuntu-20-04>

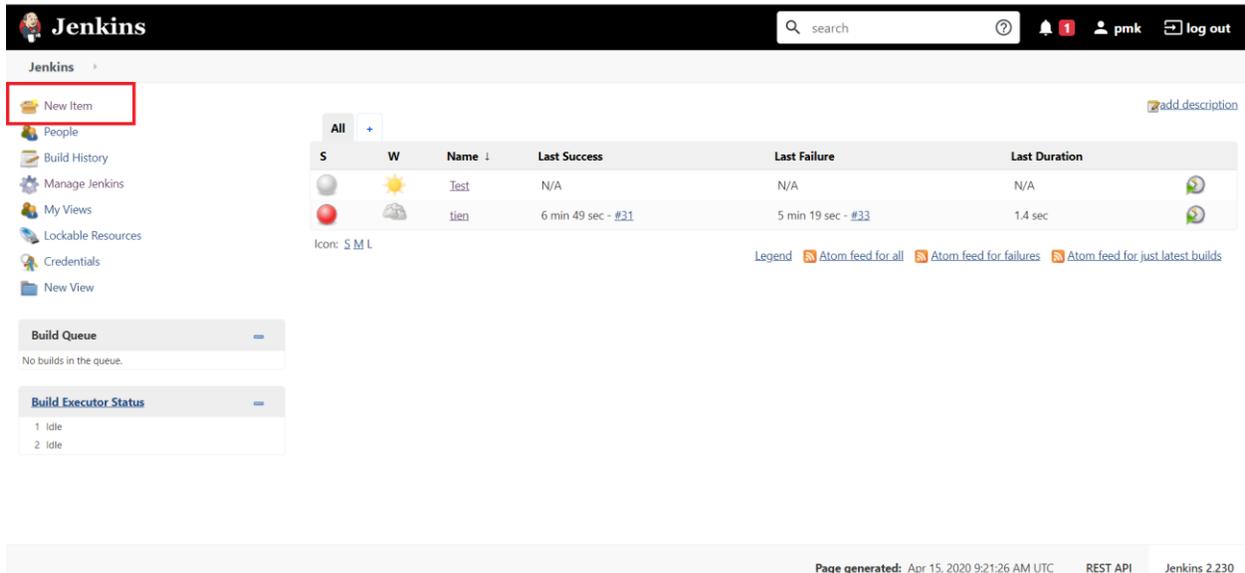
### Thực hiện Setup CI trên Jenkins

Đăng nhập vào Jenkins server. Dưới đây là màn hình chính của Jenkins server.

http:// 10.215.26.93:8080

user/pass: admin/vnpro@123

- Chọn New item để tiến hành tạo mới 1 project.



The screenshot shows the Jenkins web interface. The 'New Item' button is highlighted with a red box. Below it, there are sections for 'Build Queue' (No builds in the queue) and 'Build Executor Status' (1 Idle, 2 Idle). A table of build jobs is visible, with columns for Status, Weather, Name, Last Success, Last Failure, and Last Duration.

S	W	Name ↓	Last Success	Last Failure	Last Duration
☐	☀	test	N/A	N/A	N/A
🔴	☁	tien	6 min 49 sec - #31	5 min 19 sec - #33	1.4 sec

Ở đây chúng ta tiến hành đặt tên cho project và chọn vào mục Freestyle project. Sau đó nhấn OK.

Enter an item name

DemoCICD  
requires java

**Freestyle project**  
This is the central feature of Jenkins. Jenkins will build your project, combining any SCM with any build system, and this can be even used for something other than software build.

**Pipeline**  
Orchestrates long-running activities that can span multiple build agents. Suitable for building pipelines (formerly known as workflows) and/or organizing complex activities that do not easily fit in free-style job type.

**Multi-configuration project**  
Suitable for projects that need a large number of different configurations, such as testing on multiple environments, platform-specific builds, etc.

**Folder**  
Creates a container that stores nested items in it. Useful for grouping things together. Unlike view, which is just a filter, a folder creates a separate namespace, so you can have multiple things of the same name as long as they are in different folders.

**GitHub Organization**  
GitHub organization (or user account) for all repositories matching some defined markers.

- Ở General chúng ta tích vào Github project và nhập vào đó đường dẫn đến Git Repo dùng cho project này.

Jenkins DemoCICD search pmk log out

General Source Code Management Build Triggers Build Environment Build Post-build Actions

Description

[Plain text] Preview

Discard old builds

**GitHub project**

Project url  Advanced...

This build requires lockable resources

This project is parameterized

Throttle builds

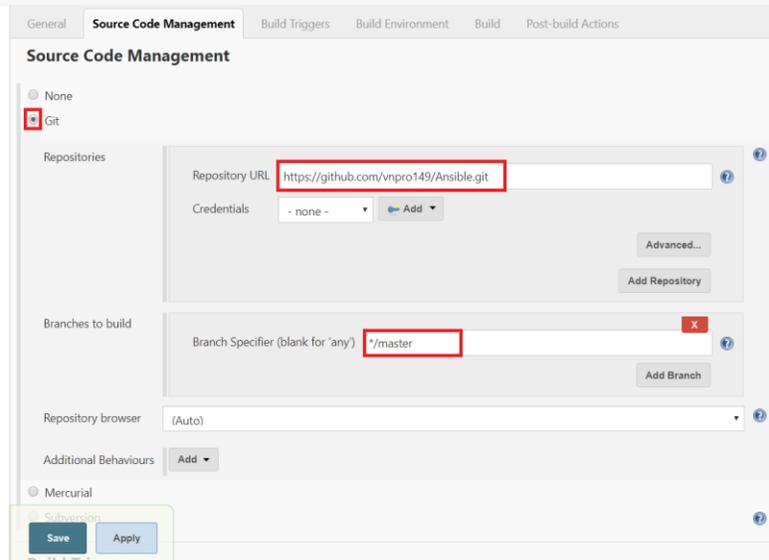
Disable this project

Execute concurrent builds if necessary

Advanced...

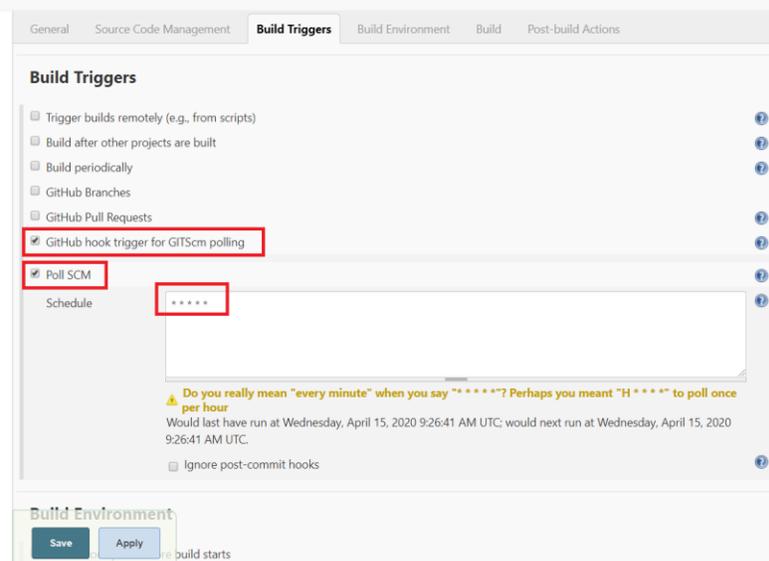
Save Apply Management

- Ở Source Code Management chúng ta tích vào ô Git, sau đó cũng nhập đường dẫn Repositories. Ở mục Branch to build chúng ta có thể tùy chọn Branch mà mình muốn sử dụng cho project, ở đây mình chọn branch master, nếu để trống thì Jenkins mặc định sẽ hiểu là any.

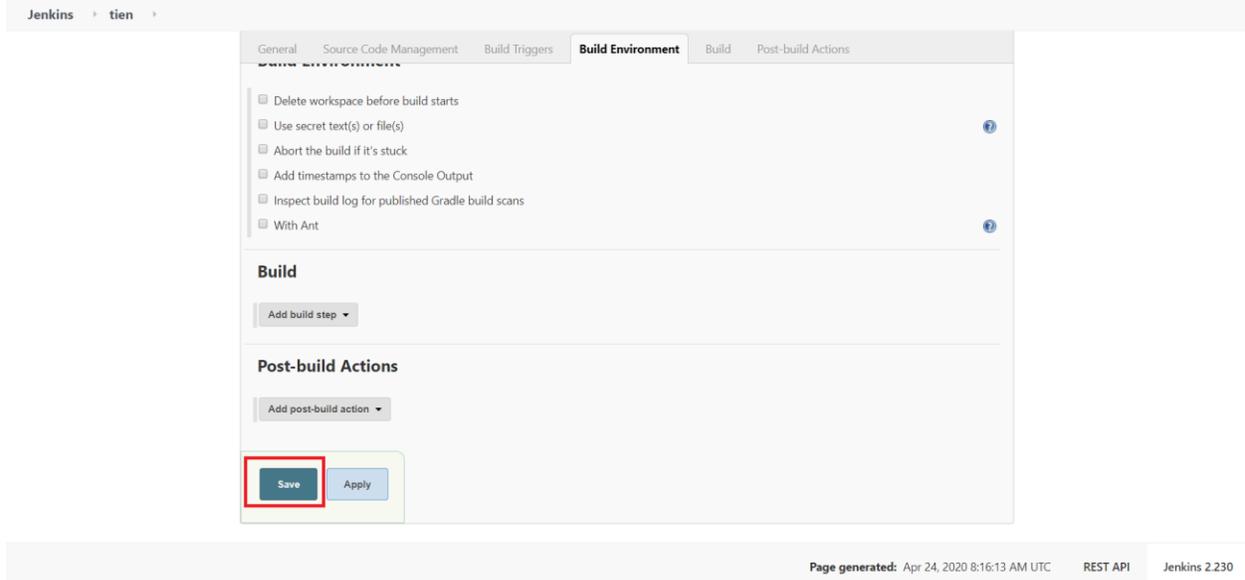


- Ở Build Triggers chúng ta tiến hành tích vào 2 ô đó là : Github hook trigger for GITScm polling và Poll SCM. Ý nghĩa của việc tích 2 ô này là khi Git Repo có sự thay đổi gì sẽ thông báo đến cho Jenkins.

**Lưu ý :** \* \* \* \* \* ở Schedule là cứ mỗi phút sẽ poll 1 lần. Giá trị này chúng ta có thể setup tùy ý.



- Tiếp theo chọn Save để lưu lại project.



- Sau khi lưu thì đây là màn hình quản lý project chúng ta vừa tạo. Có thể thấy hiện tại ở ô Build History đang trống vì project chưa build lần nào.



- Bây giờ chúng ta tiến hành tạo một file mới tại Git Repo để kiểm tra.

File successfully deleted.

Learn Git and GitHub without any code!  
Using the Hello World guide, you'll start a branch, write comments, and open a pull request.

[Read the guide](#)

vnpro149 / Ansible  
forked from hotuanhoanh/Ansible

Code Pull requests 0 Actions Projects 0 Wiki Security 0 Insights Settings

No description, website, or topics provided. [Edit](#)

Manage topics

217 commits 2 branches 0 packages 0 releases 3 contributors

Branch: master New pull request **Create new file** Upload files Find file Clone or download

- Đặt tên cho file và điền một số nội dung cho file sau đó chọn Commit để commit lại file.

Learn Git and GitHub without any code!  
Using the Hello World guide, you'll start a branch, write comments, and open a pull request.

[Read the guide](#)

vnpro149 / Ansible  
forked from hotuanhoanh/Ansible

Code Pull requests 0 Actions Projects 0 Wiki Security 0 Insights Settings

Ansible / DemoCI/CD [Cancel](#)

Edit new file Preview Spaces 2 No wrap

```
1 #DemoCI/CD
```

- Lúc này sau khi commit file xong chúng ta quay lại Jenkins server sẽ xuất hiện một thông báo có Build History. ( ở đây số 5 là vì trước đó mình đã có build một số lần khác, để dễ nhìn thì mình đã xóa các lần build trước đó )

Project DemoCICD

Build History trend ==

#5 Apr 24, 2020 8:27 AM

Atom.feed.for.all Atom.feed.for.failures

Page generated: Apr 24, 2020 8:28:27 AM UTC REST API Jenkins 2.230

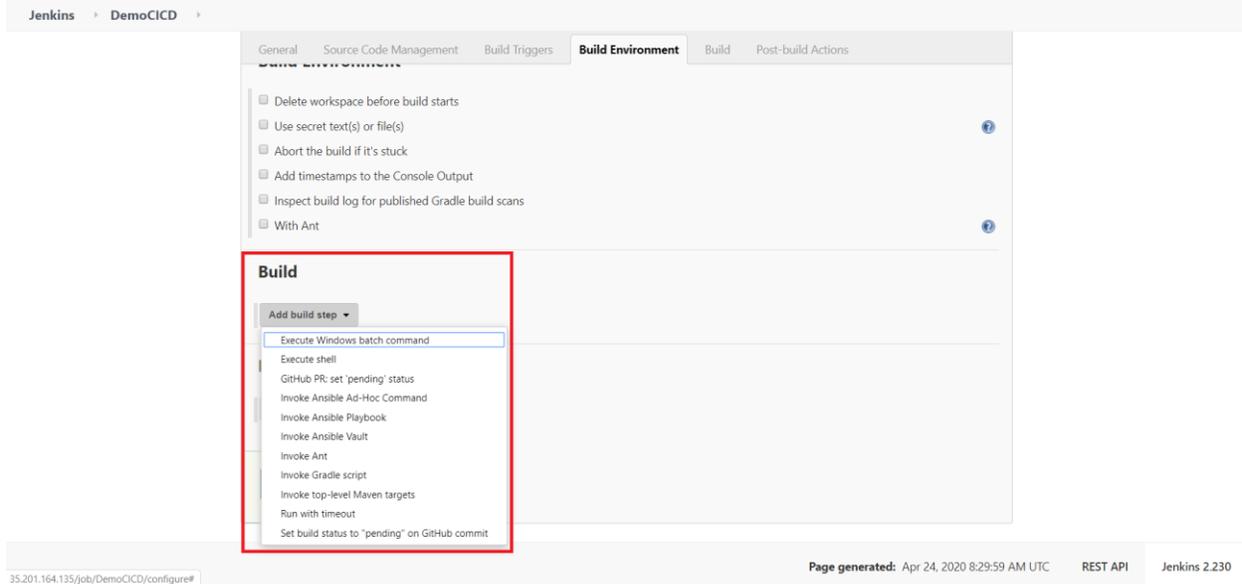
- Click vào chúng ta sẽ thấy Console Output của lần build vừa rồi. Ở đây chúng ta sẽ thấy có commit message là Create DemoCICD là file mình vừa tạo trên Git.

Console Output

```
Started by an SCM change
Running as SYSTEM
Building in workspace /var/jenkins_home/workspace/DemoCICD
No credentials specified
> /usr/bin/git rev-parse --is-inside-work-tree # timeout=10
Fetching changes from the remote Git repository
> /usr/bin/git config remote.origin.url https://github.com/vnpro149/Ansible.git # timeout=10
Fetching upstream changes from https://github.com/vnpro149/Ansible.git
> /usr/bin/git --version # timeout=10
> /usr/bin/git fetch --tags --progress -- https://github.com/vnpro149/Ansible.git +refs/heads/*:refs/remotes/origin/* # timeout=10
> /usr/bin/git rev-parse refs/remotes/origin/master^{commit} # timeout=10
> /usr/bin/git rev-parse refs/remotes/origin/master^{commit} # timeout=10
Checking out Revision 63b079384f51ceb72fd3e5eb73192b503da057e9 (refs/remotes/origin/master)
> /usr/bin/git config core.sparsecheckout # timeout=10
> /usr/bin/git checkout -f 63b079384f51ceb72fd3e5eb73192b503da057e9 # timeout=10
Commit message: "Create DemoCICD"
First time build. Skipping changelog.
Finished: SUCCESS
```

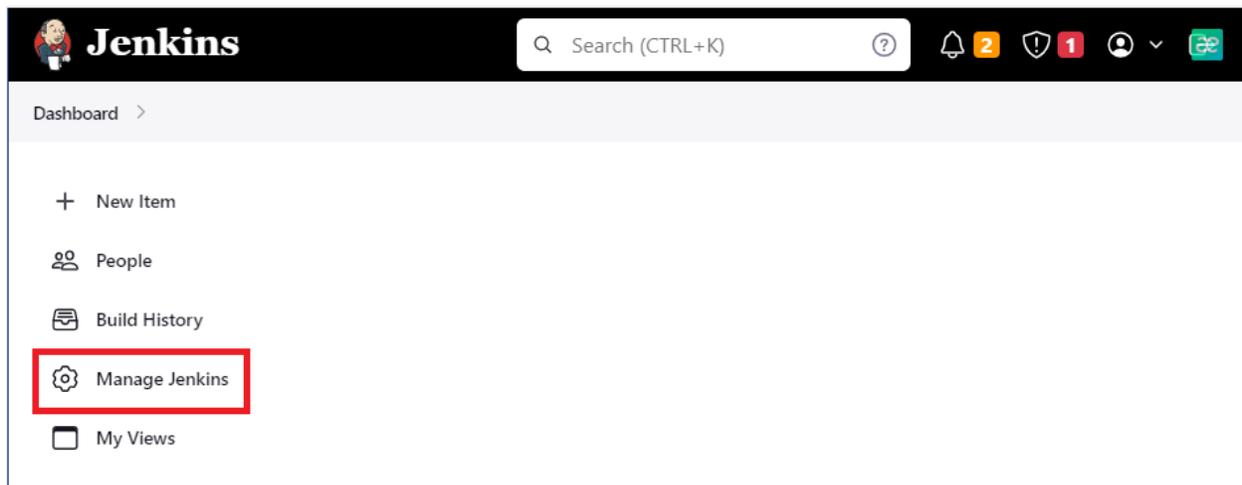
Page generated: Apr 24, 2020 8:29:19 AM UTC REST API Jenkins 2.230

**Lưu ý :** ở mục Build Enviroment chúng ta có thể chọn rất nhiều cách build khác mà Jenkins hỗ trợ như là Ansible Ad Hoc Command, Ansible Playbook, ...



- Cài đặt Ansible Tower Plugin:

Chọn Manage Jenkins:



Trong phần System Configuration chọn Manage Plugins

Dashboard > Manage Jenkins

### System Configuration

- Configure System**  
Configure global settings and paths.
- Global Tool Configuration**  
Configure tools, their locations and automatic installers.
- Manage Plugins**  
Add, remove, disable or enable plugins that can extend the functionality of Jenkins.
- Manage Nodes and Clouds**  
Add, remove, control and monitor the various nodes that Jenkins runs jobs on.

- Chọn Available plugins sau đó nhập Ansible Tower và tiến hành cài đặt

Dashboard > Manage Jenkins > Plugin Manager

- Updates
- Available plugins** 1
- Installed plugins
- Advanced settings

### Plugins

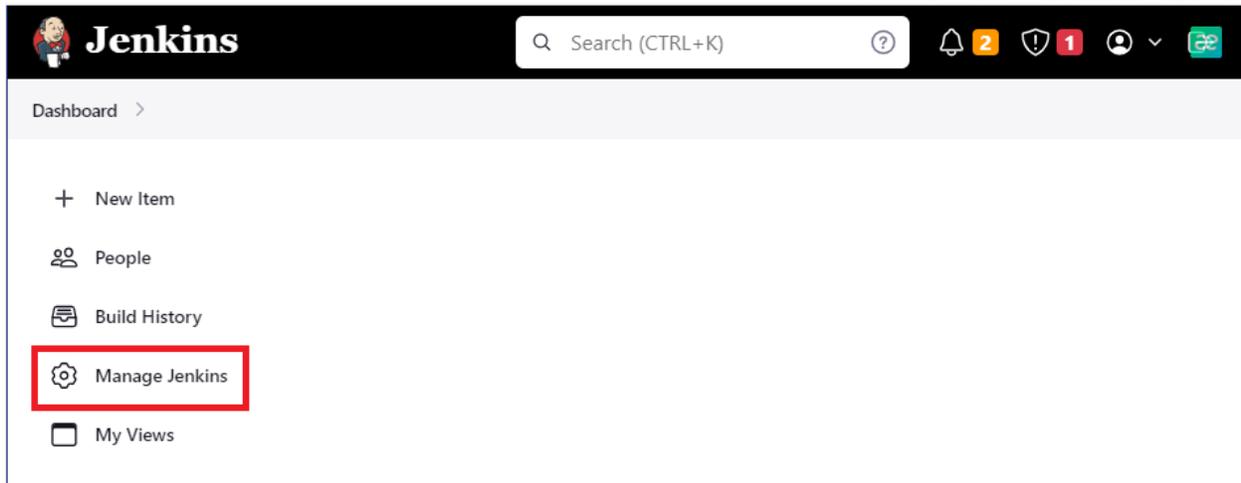
2

Install	Name ↓	Released
<input checked="" type="checkbox"/>	<b>Ansible Tower</b> 0.16.0 This plugin connects Jenkins with Ansible Tower	3 yr 0 mo ago

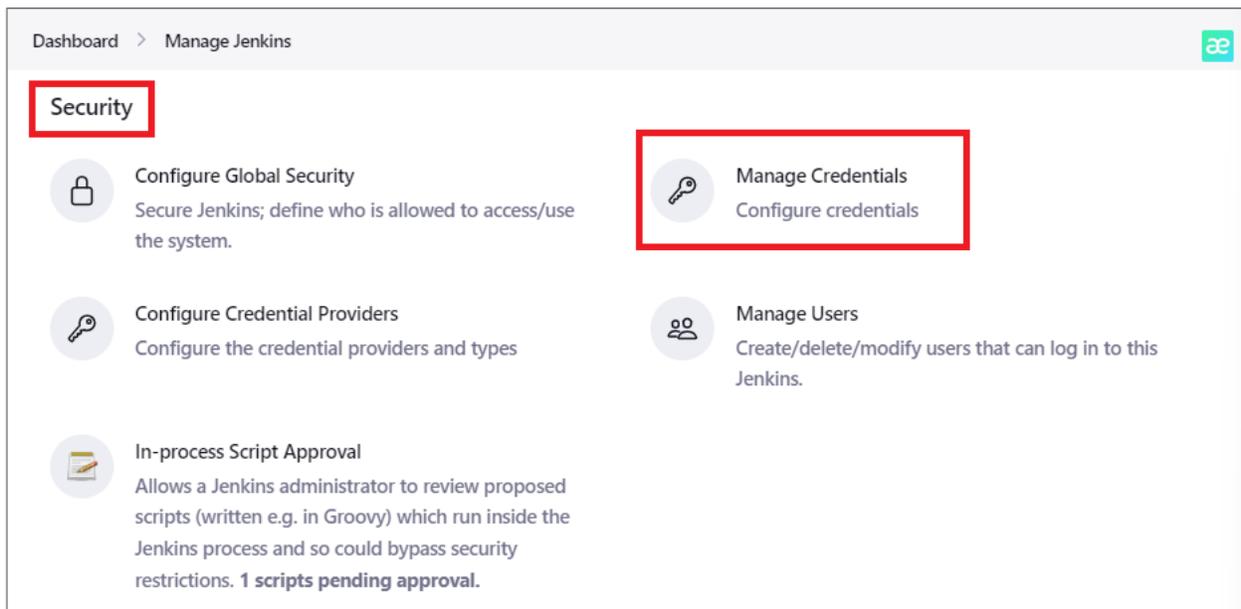
3

4  Update information obtained: 1 hr 45 min ago

- Sau khi cài đặt xong, Tiến hành cài đặt Credentials cho Ansible Tower  
Vào Manage Jenkins



Trong phần Security chọn Manage Credentials



Trong mục Stores scoped to Jenkins, click vào mục global

## Credentials

T	P	Store ↓	Domain	ID	Name
---	---	---------	--------	----	------

### Stores scoped to Jenkins

P	Store ↓	Domains
	System	(global) ↓

Icon: S M L

- Chọn Add Credentials để tạo Credential mới

The screenshot shows the Jenkins interface for managing credentials. The breadcrumb trail is: Dashboard > Manage Jenkins > Credentials > System > Global credentials (unrestricted). The main heading is "Global credentials (unrestricted)" with a blue "+ Add Credentials" button highlighted by a red box. Below the heading, there is a table with columns: ID, Name, Kind, and Description. The table is currently empty, and a message below it says: "This credential domain is empty. How about adding some credentials?". At the bottom left, there are icons for S, M, and L.

- Chọn thông số cho Credential

## New credentials

Kind

Username with password

Scope ?

Global (Jenkins, nodes, items, all child items, etc)

Username ?

admin

Treat username as secret ?

Password ?

.....

ID ?

Description ?

Create

Kind Username with password để tạo Credentials xác thực bằng username và password.

Scope để xác định phạm vi của Credential.

Username và password cho Credential này.(Credential này được sử dụng khi xác thực với Ansible Tower nên mình sẽ sử dụng username và password của Ansible Tower).

Sau khi điền đầy đủ các thông tin chọn Create để tạo Credential.

Dashboard > Manage Jenkins > Credentials > System > Global credentials (unrestricted) >

## Global credentials (unrestricted)

+ Add Credentials

Credentials that should be available irrespective of domain specification to requirements matching.

ID	Name	Kind	Description
 f8e69f2b-67fd-47d7-9843-ac35aba0b7f5	f8e69f2b-67fd-47d7-9843-ac35aba0b7f5	Username with password	

Icon: S M L

- Cấu hình Server Ansible Tower chọn Configure System

Dashboard > Manage Jenkins



### System Configuration



**Configure System**  
Configure global settings and paths.



**Global Tool Configuration**  
Configure tools, their locations and automatic installers.



**Manage Plugins**  
Add, remove, disable or enable plugins that can extend the functionality of Jenkins.



**Manage Nodes and Clouds**  
Add, remove, control and monitor the various nodes that Jenkins runs jobs on.

- Trong mục Ansible Tower cấu hình thông tin server

## Ansible Tower

### Tower Installation

Name ? ✕

URL ?

Credentials

+ Add

Force Trust Cert ?

Enable Debugging ?

Test Connection

Sau khi cấu hình xong click test connection để kiểm tra kết nối

Success

Test Connection

- Tạo mới 1 Project chọn chế độ Pipeline, cấu hình thông số tương tự như trên.
- Vào phần Pipeline chọn Pipeline script.

## Pipeline

### Definition

Pipeline script

#### Script ?

1	try sample Pipeline... ▼
---	--------------------------

Use Groovy Sandbox ?

[Pipeline Syntax](#)

Tại đây mình sẽ viết script pipeline cho project này.

Jenkins có cung cấp 1 trang hỗ trợ chúng ta viết step cho pipeline. Click vào Pipeline Syntax.

-

- Tạo Step để tương tác với Git.

## Steps

### Sample Step

git ?

Repository URL ?

Branch ?

Credentials ?

Include in polling? ?

Include in changelog? ?

**Generate Pipeline Script**

```
git 'https://github.com/vnpro149/Ansible'
```

Sample Step chọn Git để tương tác với git.

Repository URL sẽ điền đường dẫn chứa các file cấu hình sử dụng cho Ansible.

Branch để chọn Branch cần lấy file trong repository.

Sau khi điền các thông số click vào Generate Pipeline Script để tạo script cho step.

- Tạo Step tương tác với job template trên Ansible Tower

## Steps

### Sample Step

ansibleTower: Have Ansible Tower run a job template

ansibleTower

Tower Server ?

Ansible Towner

Tower Credentials ID ?

f8e69f2b-67fd-47d7-9843-ac35aba0b7f5

+ Add

Template Type ?

job

Template ID ?

DemoCD

Extra Vars ?

Tower Server chọn Ansible Tower

Tower Credentials Override chọn ID Credential

Template Type chọn job để tạo Job trên Ansible Tower

Template ID là ID hoặc tên của template đã tạo bên Ansible Tower.

Inventory ?

Credential ?

SCM Branch ?

Verbose ?

Import Tower Output ?

Import Workflow Child Output ?

Remove Color ?

Throw exception on failure ?

Run Async ?

**Generate Pipeline Script**

```
ansibleTower credential: 'DemoCD', inventory: 'DeviceCD', jobTemplate: 'DemoCD', jobType: 'run', throwExceptionWhenFail: false, towerCredentialsId: 'f8e69f2b-67fd-47d7-9843-ac35aba0b7f5', towerLogLevel: 'full', towerServer: 'Ansible Tower'
```

Inventory nhập tên Inventory được tạo bên Ansible Tower.

Credential nhập tên Credential được tạo bên Ansible Tower.

Sau khi cấu hình xong chọn Generate Pipeline Script để tạo script

- Tạo Script pipeline với 2 script step đã tạo .

```
pipeline{
  agent any
  stages{
    stage('SCM checkout'){
      steps{
        git 'https://github.com/vnpro149/Ansible'
      }
    }
    stage('Execute Ansible'){
      steps{
        ansibleTower credential: 'DemoCD', inventory: 'DeviceCD', jobTemplate:
'DemoCD', jobType: 'run', throwExceptionWhenFail: false, towerCredentialsId:
'f8e69f2b-67fd-47d7-9843-ac35aba0b7f5', towerLogLevel: 'full', towerServer: 'Ansible
Towner'
      }
    }
  }
}
```

## Pipeline

### Definition

Pipeline script

#### Script ?

```
1 pipeline{
2   agent any
3   stages{
4     stage('SCM checkout'){
5       steps{
6         git 'https://github.com/vnpro149/Ansible'
7       }
8     }
9     stage('Execute Ansible'){
10      steps{
11        ansibleTower credential: 'DemoCD', inventory: 'DeviceCD', jobTempl
12      }
13    }
14  }
15 }
16
17
```

The script is already approved

Use Groovy Sandbox ?

[Pipeline Syntax](#)

Save

Apply

Sau đó lưu lại và tiến hành thực thi.

Dashboard > DemoCICD1 >

- Status
- Changes
- Build Now**
- Configure
- Delete Pipeline
- Full Stage View
- GitHub
- Rename
- Pipeline Syntax
- GitHub Hook Log
- Git Polling Log

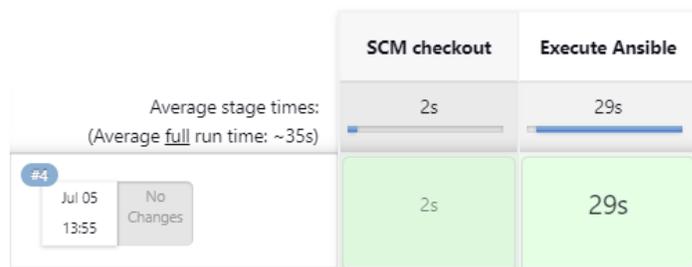
Jenkins sẽ cung cấp cho chúng ta giao diện trực quan hóa về các tiến trình đang thực thi.

## Pipeline DemoCICD1

Add description

**Disable Project**

### Stage View



Để xem chi tiết tiến trình đang thực thi click vào số id build của tiến trình đó.

Dashboard > DemoCICD1 > #4



</> Changes

☞ Console Output

✍ Edit Build Information

🗑 Delete build '#4'

🔴 Git Build Data

🔄 Restart from Stage

↺ Replay

☰ Pipeline Steps

📁 Workspaces

← Previous Build

## ✔ Build #4 (Jul 5, 2023, 1:55:02 PM)

Keep this build forever

✎ Add description

Started 31 min ago

Took 35 sec



Started by user [trongchien](#)



Revision: 3c90ea57b8b32ec036a8e9f95eb9ad384add2c1d

Repository: <https://github.com/vnpro149/Ansible>

- refs/remotes/origin/master

Để xem chi tiết hơn Click vào Console Output.



```
[Pipeline] { (Execute Ansible)
[Pipeline] ansibleTower
[WARNING]: Job type defined but prompt for job type on launch is not set in tower job
[WARNING]: Inventory defined but prompt for inventory on launch is not set in tower job
[WARNING]: Credential defined but prompt for credential on launch is not set in tower job
Template Job URL: http://10.215.26.65/#/jobs/1674
SSH password:
@[0;35m[DEPRECATION WARNING]: ios_vlan is kept for backwards compatibility but usage @[0m
@[0;35mis discouraged. The module documentation details page may explain more about @[0m
@[0;35mthis rationale.. This feature will be removed in a future release. Deprecation @[0m
@[0;35mwarnings can be disabled by setting deprecation_warnings=False in ansible.cfg.@[0m

PLAY [demo] *****

TASK [Gathering Facts] *****

@[1;35m[WARNING]: Ignoring timeout(10) for ios_facts@[0m
@[1;35m[WARNING]: default value for `gather_subset` will be changed to `min` from@[0m
@[1;35m`!config` v2.11 onwards@[0m
@[0;32mok: [Switch]@[0m

TASK [Create vlan] *****

@[0;33mchanged: [Switch]@[0m

PLAY RECAP *****
@[0;33mSwitch@[0m          : @[0;32mok=2  @[0m @[0;33mchanged=1  @[0m unreachable=0  failed=0
skipped=0  rescued=0  ignored=0
Tower completed the requested job
[Pipeline] }
[Pipeline] // stage
[Pipeline] }
[Pipeline] // node
[Pipeline] End of Pipeline
Finished: SUCCESS
```