

CD project with Jenkins Nexus and with Ansible :

We need Jenkins, GIT, Maven, Nexus and Ansible Installation and need 5 EC2 Instances to run this project.

1. EC2 Server (Ubuntu)

We have to install Jenkins, GIT, Maven, Ansible here as follows.

Jenkins Setup On Ubuntu Server :

Jenkins can be installed on windows, Linux or Mac OS. Jenkins just needs java software to run.

In this tutorial, we will install jenkins on a ubuntu server. You can setup a vm or a cloud instance. Prereqs Java runtime environment/ JRE can be installed on the system but we will install JDK as we will setup maven moving along and build some java code. To Build the java code we will need JDK.

```
sudo add-apt-repository ppa:openjdk-r/ppa
```

```
sudo apt-get update
```

```
sudo apt-get install openjdk-8-jdk
```

Installing Jenkins on Ubuntu :

```
wget -q -O - https://pkg.jenkins.io/debian/jenkins-ci.org.key | sudo apt-key add -  
sudo sh -c 'echo deb http://pkg.jenkins.io/debian-stable binary/ > /etc/apt/sources.list.d/jenkins.list'  
sudo apt-get update  
sudo apt-get install jenkins
```

Install git client and maven in jenkins server

We will integrate Jenkins with github to download the source code.

We are testing the java source code which will be built by Maven, so we also need to install Maven on Jenkins server. This is not a mandatory requirement to run Jenkins if you are not using git and maven.

```
sudo apt-get install git
```

```
sudo apt-get install maven
```

Install Ansible in jenkins server (Ubuntu) :

- sudo apt-get update
- sudo apt-get install software-properties-common
- sudo apt-add-repository ppa:ansible/ansible
- sudo apt-get update
- sudo apt-get install ansible

Jenkins Server Security Group setup :

Edit inbound rules ✕

Type i	Protocol i	Port Range i	Source i	Description i	
Custom TCP f	TCP	8080	Custom v 183.82.230.169/32	e.g. SSH for Admin Desktop	✕
SSH v	TCP	22	Custom v 183.82.230.169/32	e.g. SSH for Admin Desktop	✕

Add Rule

NOTE: Any edits made on existing rules will result in the edited rule being deleted and a new rule created with the new details. This will cause traffic that depends on that rule to be dropped for a very brief period of time until the new rule can be created.

Cancel Save

SSH ----- Custom ----- MyIP

Custom TCP Protocol --- 8080 ----- Custom ----- MyIP

2. Nexus Server setup (Centos/6)

- ❖ `sudo -i`
- ❖ `yum search java`
- ❖ Here we can identify all the versions of java.
- ❖ `yum install -y java-1.8.0-openjdk.x86_64 vim wget`
- ❖ `export RUN_AS_USER=root`
- ❖ `wget http://www.sonatype.org/downloads/nexus-latest-bundle.tar.gz`
- ❖ `sudo cp nexus-latest-bundle.tar.gz /usr/local/`
- ❖ `cd /usr/local`
- ❖ `sudo tar -xvzf nexus-latest-bundle.tar.gz`
- ❖ `sudo ln -s <nexus directory name> nexus`
- ❖ `cd nexus`
- ❖ `cd bin`
- ❖ `./nexus start`
- ❖ `Export RUN_AS_USER=root`
- ❖ `./nexus start`
- ❖ `Service iptables status`
- ❖ `service iptables stop`
- ❖ And we have to enable traffic from Jenkins security group in nexus security group from 8081.
- ❖ open browser take IP address and port number 8081 and `/nexus`
- ❖ `http://13.57.36.214:8081/nexus`
- ❖ log in user name and password ----- admin admin123

Accessing Nexus dashboard :

- ✓ From browser hit URL <Nexus server IP>:8081/nexus.
- ✓ Click login button and enter the credentials. (admin/admin123) .
- ✓ Click on Repositories and click on Add and select Hosted repository.

The screenshot shows the Nexus Repository Manager OSS interface. The browser address bar displays the URL: 18.144.24.170:8081/nexus/#view-repositories;VProfile-repo~browsestorage. The left sidebar contains navigation links: Sonatype™, Artifact Search, Views/Repositories (with 'Repositories' highlighted), Security, Administration, and Help. The main content area shows the 'Repositories' tab with a table of existing repositories. The 'VProfile-repo' repository is highlighted. Below the table, the 'VProfile-repo' details are shown, including a 'Browse Storage' button and a 'Path Lookup' section.

Repository	Type	Health Check	Format	Policy
Central	proxy	ANALYZE	maven2	Rel
Central M1 shadow	virtual	ANALYZE	maven1	Rel
Releases	hosted	ANALYZE	maven2	Rel
Snapshots	hosted	ANALYZE	maven2	Sna
vprofile-repo	hosted	ANALYZE	maven2	Rel
VProfile-repo	hosted	ANALYZE	maven2	Rel

VProfile-repo

Browse Index | **Browse Storage** | Configuration | Routing | Summary | Artifact Upload

Refresh Path Lookup: [x] [p]

VProfile-repo

- Dev1
- QA

- ✓ Create hosted repository named “VProfile-repo” with all default settings. And then we have to create job in Jenkins.

Nexus Server Security Group setup :

Edit inbound rules

Type	Protocol	Port Range	Source	Description	
SSH	TCP	22	Custom 0.0.0.0/0	allows from tomcat	✕
SSH	TCP	22	Custom 183.82.230.169/32	allows from my ip	✕
SSH	TCP	22	Custom ::/0	allows from tomcat	✕
Custom TCP f	TCP	8081	Custom 183.82.230.169/32	Allows from my ip	✕
Custom TCP f	TCP	8081	Custom sg-01a4490014f638eb4	allows from jenkins	✕
Custom TCP f	TCP	8081	Custom sg-07c2fd0a354cd2d7c	Allows from Tomcat Servers	✕

Add Rule

NOTE: Any edits made on existing rules will result in the edited rule being deleted and a new rule created with the new details. This will cause traffic that depends on that rule to be dropped for a very brief period of time until the new rule can be created.

Cancel Save

- Remaining 3 servers we have to just create EC2 Instances type is centos/6.
Launch Instance and type centos/6 and select 3 and setup security group, that's it no need to install anything here.

Tomcat Servers security group setup

Edit inbound rules

Type	Protocol	Port Range	Source	Description	
Custom TCP f	TCP	8080	Custom 0.0.0.0/0	e.g. SSH for Admin Desktop	✕
Custom TCP f	TCP	8080	Custom ::/0	e.g. SSH for Admin Desktop	✕
SSH	TCP	22	Custom 0.0.0.0/0	e.g. SSH for Admin Desktop	✕
SSH	TCP	22	Custom 183.82.230.169/32	e.g. SSH for Admin Desktop	✕
SSH	TCP	22	Custom ::/0	e.g. SSH for Admin Desktop	✕

Add Rule

NOTE: Any edits made on existing rules will result in the edited rule being deleted and a new rule created with the new details. This will cause traffic that depends on that rule to be dropped for a very brief period of time until the new rule can be created.

Cancel Save

Access Jenkins from browser : < Public IP address > : 8080

The screenshot shows the Jenkins web interface. The top navigation bar includes the Jenkins logo and a search bar. The left sidebar contains links to 'New Item', 'People', 'Build History', 'Manage Jenkins', 'My Views', 'Credentials', and 'New View'. Below these are sections for 'Build Queue' (showing 'No builds in the queue') and 'Build Executor Status' (showing 2 idle executors). The main content area displays a table of jobs, with a filter set to 'VProfile-Pipeline'. The table has columns for status (S), weather icon (W), name, last success, and last failure.

S	W	Name ↓	Last Success	Last Failure
		maven_project	5 hr 32 min - #6	5 hr 32 min - #5
		Myfirstjob	6 hr 15 min - #2	N/A
		smallproject	5 hr 54 min - #19	N/A
		Vprofile-Deploy-Dev	59 min - #25	15 hr - #18
		Vprofile-Deploy-QA	58 min - #4	1 hr 20 min - #2
		Vprofile-Deploy-UAT	1 hr 27 min - #1	N/A
		Vprofile-Deploy-UAT-Pipeline	57 min - #1	N/A
		vprofile-install	19 hr - #3	18 hr - #8

Icon: [S](#) [M](#) [L](#) [Legend](#) [RSS for all](#) [RSS fo](#)

- ❖ We have to create 3 Jobs for our project to deploy into Dev - - QA - - UAT all 3 environments (Like Continuous Delivery)
- ❖ We have to install required plug ins as follows
 - ✓ Nexus artifact uploader
 - ✓ ZenTimestamp plugin
 - ✓ Ansible plugin
 - ✓ Build pipeline plgin
 - ✓ Parameterised trigger other build remotely

1. VProfile-Deploy-Dev job

General Tab :

Jenkins search

Jenkins > Vprofile-Deploy-Dev >

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

Project name

Description

[Plain text] [Preview](#)

☐ Change date pattern for the BUILD_TIMESTAMP (build timestamp) variable

☐ Discard old builds

☐ GitHub project

☐ This project is parameterized

☐ Throttle builds

☐ Disable this project

☐ Execute concurrent builds if necessary

Advanced...

Source Code Management :

Jenkins > Vprofile-Deploy-Dev >

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

Source Code Management

☐ None
☒ Git

Repositories

Repository URL

Credentials

- none -

[Add](#)

Advanced...

Add Repository

Branches to build

Branch Specifier (blank for 'any')

Add Branch

Repository browser

(Auto)

Additional Behaviours

Add

Build Triggers and Build Environment are just keep it as empty.

Build Tab :

First Build step we have to select “**Invoke top-level Maven targets**” and select Goal as “**Install**”



Second Build step we have to select “**Nexus artifact uploader**” and need to configure as follows.

Here Nexus2 is a version we have to select and Protocol as HTTP and URL should like below, IP address should be private since all are in same network like same vpc, credentials admin/admin123 and Group ID can be Project folder can be given Dev, QA like this. Remaining also just as follows.

Nexus artifact uploader

Nexus Details

Nexus Version

NEXUS2

Protocol

HTTP

Nexus URL

172.31.2.81:8081/nexus

Credentials

admin/***** (Nexus credentials)

Ad+

GroupId

Dev1

Version

\$BUILD_ID

Repository

VProfile-repo

Artifacts

Artifact

Artifacts

Artifact

ArtifactId

\$BUILD_TIMESTAMP

Type

war

Classifier

File

target/vprofile-v1.war

Add

Third Build step we have to select “**Ansible Invoke Playbook**”

Here we have to our host inventory as Inline content like below.

There is group has been here name as tomcatserver this group has been mentioned in playbook, you can verify playbook in our source code under Ansible folder and click on view option.

The image shows two screenshots from the Jenkins interface. The top screenshot displays the workspace for 'Vprofile-Deploy-Dev' on the master. It shows a folder named 'ansible' containing a file 'vprofile.yml' (1.10 KB). A red box highlights the 'view' link next to the file. The bottom screenshot shows the 'Invoke Ansible Playbook' configuration step. The 'Playbook path' is set to 'ansible/vprofile.yml'. Under the 'Inventory' section, 'Inline content' is selected. The 'Content' field contains the following text: 'devtom ansible_ssh_host=172.31.2.107 ansible_ssh_user=centos' followed by a new line and '[tomcatserver] devtom'. The 'Host subset' field is empty.

Jenkins > Vprofile-Deploy-Dev >

Back to Dashboard
Status
Changes
Workspace
Build Now
Delete Project

Workspace of Vprofile-Deploy-Dev on master

ansible /

vprofile.yml 1.10 KB [view](#)

(all files in zip)

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

Invoke Ansible Playbook

Playbook path:

Inventory:

- ☐ Do not specify Inventory
- ☐ File or host list
- ☒ Inline content

Dynamic inventory: ☐

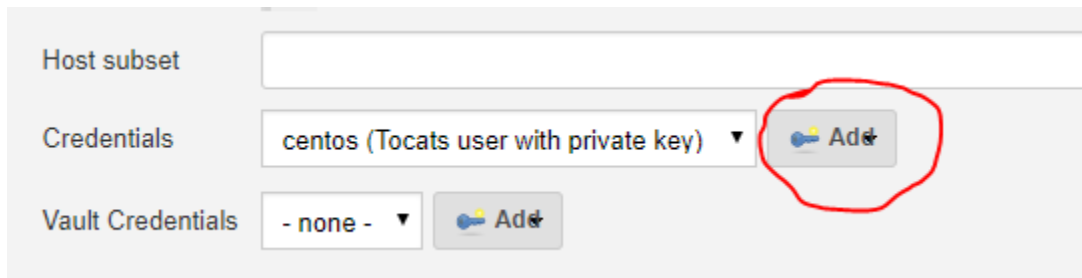
Content:

Host subset:

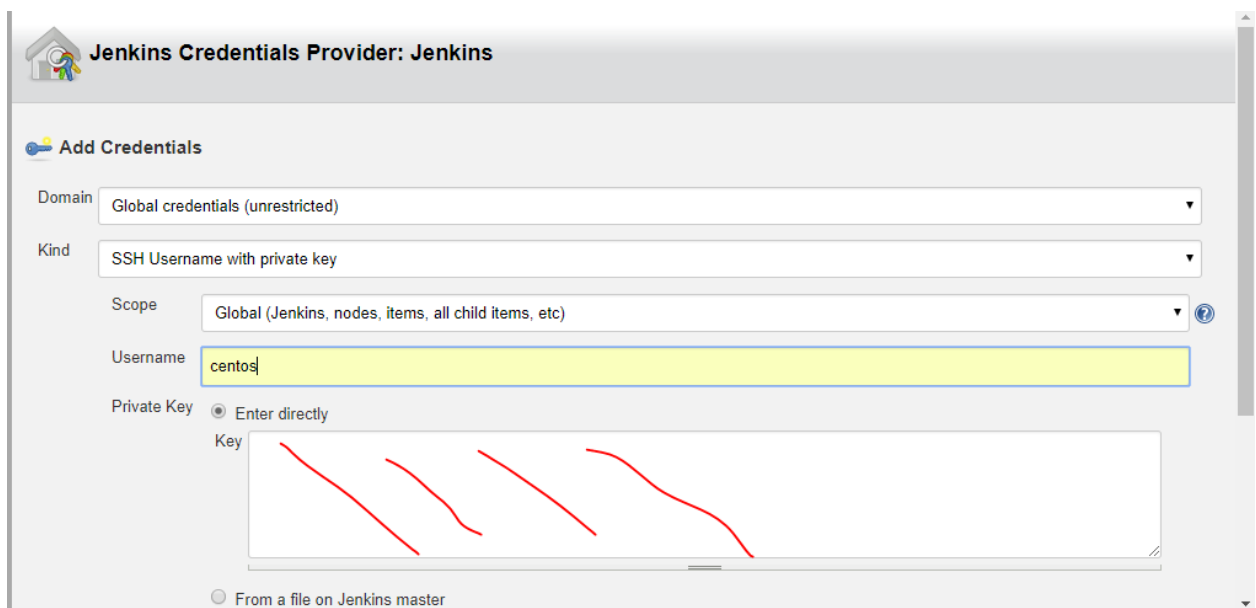
Here we have to enter dev tomcat host name as your wish and enter dev tomcat private ip and enter user name as centos since these are cnetos ec2 systems and playgroup we have to create

After adding host inventory as per yml file then we have to authenticate deployment tomcat server using user name and private key option like below.

Under Credentials Tab we have to click on Add button and select Jenkins



This screenshot shows the 'Host subset' configuration form in Jenkins. It includes three sections: 'Host subset' with an empty text field, 'Credentials' with a dropdown menu showing 'centos (Tocats user with private key)' and a red circle around the 'Add' button, and 'Vault Credentials' with a dropdown menu showing '- none -' and an 'Add' button.



This screenshot shows the 'Jenkins Credentials Provider: Jenkins' form. It includes a header with a house icon and the title 'Jenkins Credentials Provider: Jenkins'. Below the header is a section titled 'Add Credentials' with a key icon. The form contains several fields: 'Domain' (Global credentials (unrestricted)), 'Kind' (SSH Username with private key), 'Scope' (Global (Jenkins, nodes, items, all child items, etc)), 'Username' (centos), and 'Private Key' (Enter directly). The 'Private Key' section has a large text area for the key and a radio button for 'From a file on Jenkins master'.

So we just installed centos servers as tomcat servers so we gave user name as centos and select private key as Enter directly and paste the content of a Tomcat servers pem file. So go to Download folder and open out tomcat server pem file using cat command and copy and paste here and give some description and click add button. Once added credentials to our Jenkins we have to select same created credentials like below.

And finally we are declaring and providing all variable values to our playbook

```

---
- hosts: tomcatservers
  become: yes
  gather_facts: False
  tasks:

  - name: Install EPEL-release
    yum: name=epel-release state=present

  - name: Install java_1.8
    yum: name=java-1.8.0-openjdk.x86_64 state=present

  - name: Install tomcat
    yum: name=tomcat state=present

  - name: Download latest VProfile.war file
    get_url: url=http://{{nexusip}}:8081/nexus/content/repositories/VProfile-repo/{{groupid}}/{{time}}/{{build}}/{{vprofile_version}} dest=/tmp/ mode=755

  - name: Stop tomcat service
    service: name=tomcat state=stopped

  - name: Copy artifact to tomcat folder
    shell: cp /tmp/{{vprofile_version}} /var/lib/tomcat/webapps

```

General	Source Code Management	Build Triggers	Build Environment	Build	Post-build Actions
Host subset		<input type="text"/>			
Credentials		centos (Tocats user with private key)			
Vault Credentials		- none -			
<input type="checkbox"/> become					
<input type="checkbox"/> sudo					
Tags to run		<input type="text"/>			
Tags to skip		<input type="text"/>			
Task to start at		<input type="text"/>			
Number of parallel processes		<input type="text" value="5"/>			
Disable the host SSH key check		<input type="checkbox"/>			
Unbuffered stdout		<input checked="" type="checkbox"/>			
Colorized stdout		<input type="checkbox"/>			
Extra Variables		<div></div>			

Nexus ip should be private IP address

GeneralSource Code ManagementBuild TriggersBuild EnvironmentBuildPost-build Actions

Extra Variables

X

Keynexusip

Value172.31.2.81

Hidden variable in build log☐

X

Keygroupid

ValueDev1

Hidden variable in build log☐

X

Keytime

Value\$BUILD_TIMESTAMP

Hidden variable in build log☐

X

Keybuild

Value\$BUILD_ID

Hidden variable in build log☐

X

Keyvprofile_version

Value\$BUILD_TIMESTAMP-\$BUILD_ID.war

Hidden variable in build log☐

Add Extra Variable

And Click on Save finally and run the Build. So here we can see artifact has been created and uploaded into Nexus Repository and deployed into Dev Tomcat server. Build should say success and we can verify both the things like below.

```
TASK [wait_for] *****
ok: [devtom]

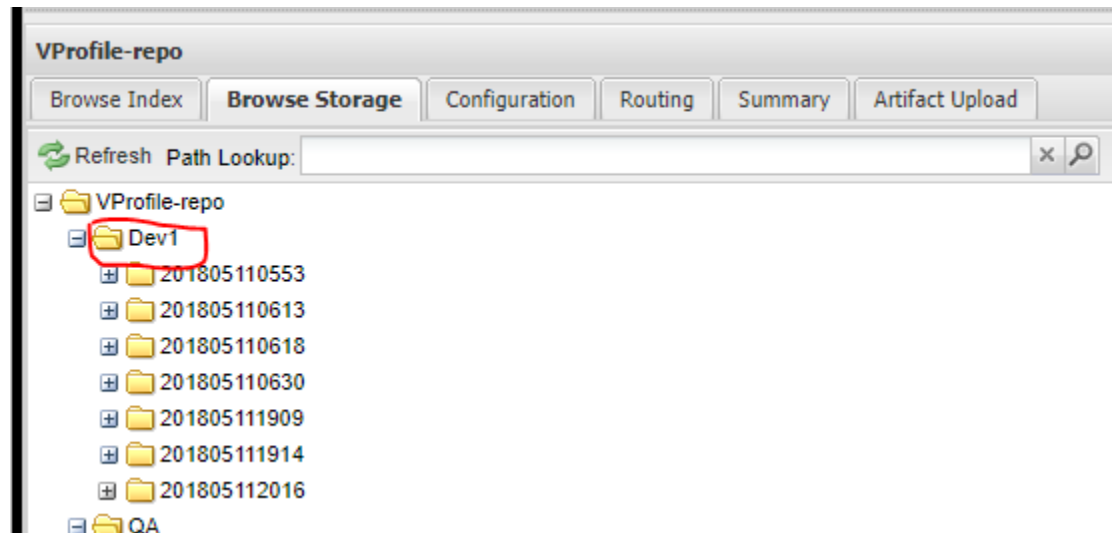
TASK [Link latest vprofile version] *****
changed: [devtom]

TASK [Stop iptables] *****
ok: [devtom]

PLAY RECAP *****
devtom                : ok=11   changed=6   unreachable=0   failed=0

Finished: SUCCESS
```

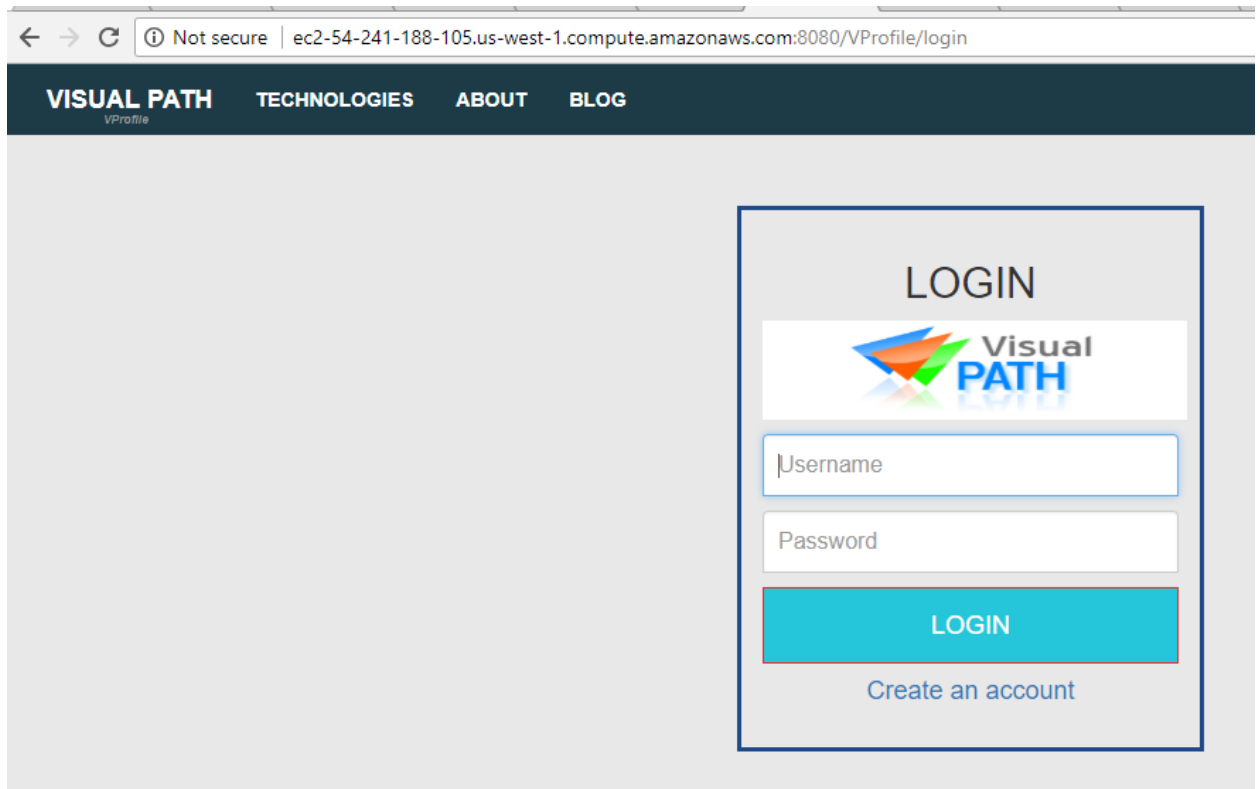
Nexus :



This can be project Name or environment name like “Dev”

And now Tomcat we have to give EC2 instance public DNS name :8080/VProfile and enter

Output like below.



Now we just setup our first build deployed to Dev Environment.

2. VProfile-Deploy-QA job : Same configuration we have to copy from previous job like below needed few changes as follows.

When we create second job just say free style and copy from previous job and configure as we required.

General Tab:

Vprofile-Deploy-QA ▾ ▸

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

Project name

Description

[Plain text] [Preview](#)

- ☐ Change date pattern for the BUILD_TIMESTAMP (build timestamp) variable ?
- ☐ Discard old builds ?
- ☐ GitHub project ?
- ☐ This project is parameterized ?
- ☐ Throttle builds ?
- ☐ Disable this project ?
- ☐ Execute concurrent builds if necessary ?

Source code management Tab :

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

Source Code Management

☐ None
☒ Git

Repositories

Repository URL ?

Credentials Add ?

Advanced...
Add Repository

Branches to build

Branch Specifier (blank for 'any') ?

Add Branch

Repository browser ?

Additional Behaviours

Build Triggers and Build Environment are just keep it as empty.

Build Tab :

General

Source Code Management

Build Triggers

Build Environment

Build

Post-build Actions

Build

Invoke top-level Maven targets

X

?

Goals

install

▼

Advanced...

Nexus artifact uploader

Nexus Details

Nexus Version

NEXUS2

Protocol

HTTP

Nexus URL

172.31.2.81:8081/nexus

Credentials

admin/***** (Nexus credentials)

Ad

GroupId

QA

Version

\$BUILD_ID

Repository

VProfile-repo

Artifacts

Artifact

Artifacts

Artifact

ArtifactId

\$BUILD_TIMESTAMP

Type

war

Classifier

File

target/vprofile-v1.war

Add

General

Source Code Management

Build Triggers

Build Environment

Build

Post-build Actions

Invoke Ansible Playbook

Playbook path

ansible/vprofile.yml

Inventory

Do not specify Inventory

File or host list

Inline content



















Dynamic inventory

Content

qatom ansible_ssh_host=172.31.9.160 ansible_ssh_user=centos
[tomcatservers]
qatom

Host subset

Here we have to enter QA tomcat host name as your wish and enter QA tomcat private ip and enter user name as centos since these are centos ec2 systems and playgroup we have to create

Host subset	<input type="text"/>	
Credentials	<div>centos (Tocats user with private key) </div> <div> Add </div>	
Vault Credentials	<div>- none - </div> <div> Add </div>	
<input type="checkbox"/> become		
<input type="checkbox"/> sudo		
Tags to run	<input type="text"/>	
Tags to skip	<input type="text"/>	
Task to start at	<input type="text"/>	
Number of parallel processes	<input type="text" value="5"/>	
Disable the host SSH key check	<input type="checkbox"/>	
Unbuffered stdout	<input checked="" type="checkbox"/>	
Colorized stdout	<input type="checkbox"/>	

The screenshot displays the Jenkins configuration interface, specifically the 'Extra Variables' and 'Additional parameters' sections. The 'Extra Variables' section contains three entries, each with a 'Key', 'Value', and a checkbox for 'Hidden variable in build log'. The first entry has Key 'nexusip' and Value '172.31.2.81'. The second entry has Key 'groupid' and Value 'QA', with the 'QA' value circled in red. The third entry has Key 'time' and Value '\$BUILD_TIMESTAMP'. The 'Additional parameters' section at the bottom has a single text input field. Each entry in the 'Extra Variables' section has a red 'X' button in the top right corner. A blue question mark icon is visible in the top right corner of the 'Extra Variables' section and the bottom right corner of the 'Additional parameters' section.

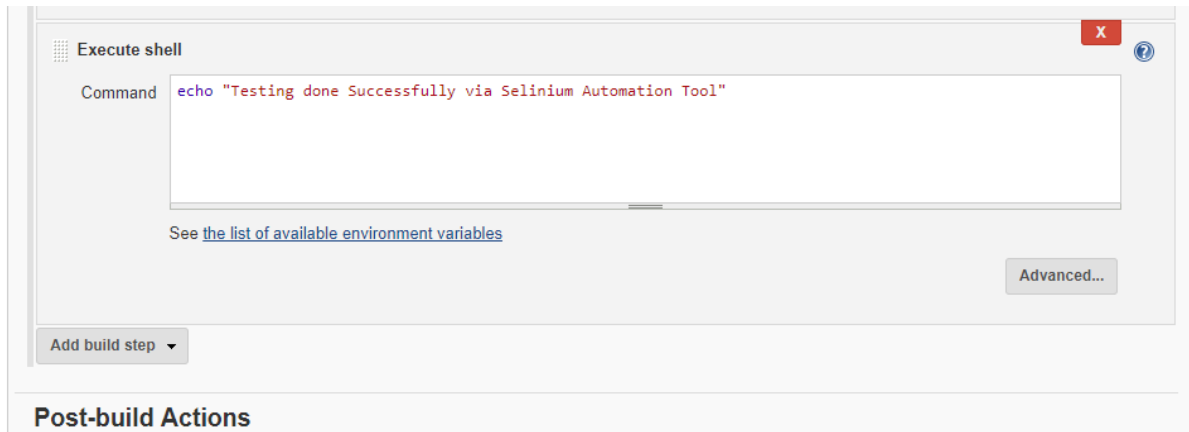
Key	Value	Hidden variable in build log
nexusip	172.31.2.81	<input type="checkbox"/>
groupid	QA	<input type="checkbox"/>
time	\$BUILD_TIMESTAMP	<input type="checkbox"/>

Additional parameters

And Click on Save finally and run the Build. So here we can see artifact has been created and uploaded into Nexus Repository and deployed into QA Tomcat server.

Build should say success and we can verify both the things like above screen (please verify checks we done for first job above). Then it is success.

And here we can add one more task under build job to make automation test against this application deployment using some selenium automation plug in, but for testing am just giving sample task like below to confirm testing is passed here so that we can move build into other environment like UAT.



3. VProfile-Deploy-UAT job : Same configuration we have to copy from previous job like below needed few changes as follows.

- ❖ When we create this job just say free style and copy from previous job and configure as we required.
- ❖ Here what we are going to do is like previous job we are not going to build from the git hub and we are not going to take artifact from nexus after deploy into nexus. So here am going to get artifact from QA env which was passed all test cases in QA.
- ❖ So build config we have to change as per our requirement.
- ❖ This project should allow to give input from user so this build can be parameterized
- ❖ We have to configure same like as follows.

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

Project name

Vprofile-Deploy-UAT

Description

[Plain text] [Preview](#)

☐ Change date pattern for the BUILD_TIMESTAMP (build timestamp) variable

☐ Discard old builds

☐ GitHub project

?

?

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

☒ This project is parameterized ?

String Parameter X ?

Name

Time

Default Value

Description

[Plain text] [Preview](#)

☐ Trim the string ?

String Parameter X ?

Name

ID

Default Value

Description

[Plain text] [Preview](#)

Save

Apply

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

String Parameter

Name

Default Value

Description

[Plain text] [Preview](#)

☐ Trim the string

Add Parameter

☐ Throttle builds

☐ Disable this project

☐ Execute concurrent builds if necessary

Advanced...

Actually in this we are not going to run build we are taking artifact from QA which was tested and passed, but we need playbook to deploy that is why we need this github url.

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

Source Code Management

☐ None

☒ Git

Repositories

Repository URL

Credentials [Add](#)

Advanced...

Add Repository

Branches to build

Branch Specifier (blank for 'any')

Add Branch

Repository browser

Additional Behaviours [Add](#)

Subversion

Build Triggers and Build Environment are just keep it as empty.

Build Tab :

Build

Invoke top-level Maven targets

Goals

Advanced...

Nexus artifact uploader

GeneralSource Code ManagementBuild TriggersBuild EnvironmentBuildPost-build Actions

Nexus artifact uploader

Nexus Details

Nexus VersionNEXUS2

ProtocolHTTP

Nexus URL172.31.2.81:8081/nexus

Credentialsadmin/***** (Nexus credentials) Add

GroupIdQA same QA

Version\$BUILD_ID

RepositoryVProfile-repo

Artifacts

Artifacts

Artifact

ArtifactId\$BUILD_TIMESTAMP

Typewar

Classifier

Filetarget/vprofile-v1.war

Add

GeneralSource Code ManagementBuild TriggersBuild EnvironmentBuildPost-build Actions

Invoke Ansible Playbook

Playbook path

ansible/vprofile.yml

Inventory

Do not specify Inventory

File or host list

Inline content

Dynamic inventory

Content

uatom ansible_ssh_host=172.31.8.255 ansible_ssh_user=centos

[tomcatserverns]

uatom

Host subset

Credentials

centos (Tocats user with private key)

Vault Credentials

- none -

Vault Credentials

- none -

become

sudo

Tags to run

Tags to skip

Task to start at

Number of parallel processes

Disable the host SSH key check

Unbuffered stdout

Colorized stdout

Extra Variables

5

Key

nexusip

Value

172.31.2.81

Hidden variable in build log

Vprofile-Deploy-UAT

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

Key: groupid
Value: QA
Hidden variable in build log: ☐

Key: time
Value: \$Time
Hidden variable in build log: ☐

Key: build
Value: \$ID
Hidden variable in build log: ☐

Key: vprofile_version
Value: \$Version.war
Hidden variable in build log: ☐

Save Apply

The values are giving by the users when we run this job those values can be used here to run this job. Click on Save and now Say Build with Parameters.

These values can be taken from QA Group id from Nexus repository like below,

Jenkins

Jenkins > Vprofile-Deploy-UAT

Back to Dashboard
Status
Changes
Workspace
Build with Parameters
Delete Project
Configure

Project Vprofile-Deploy-UAT

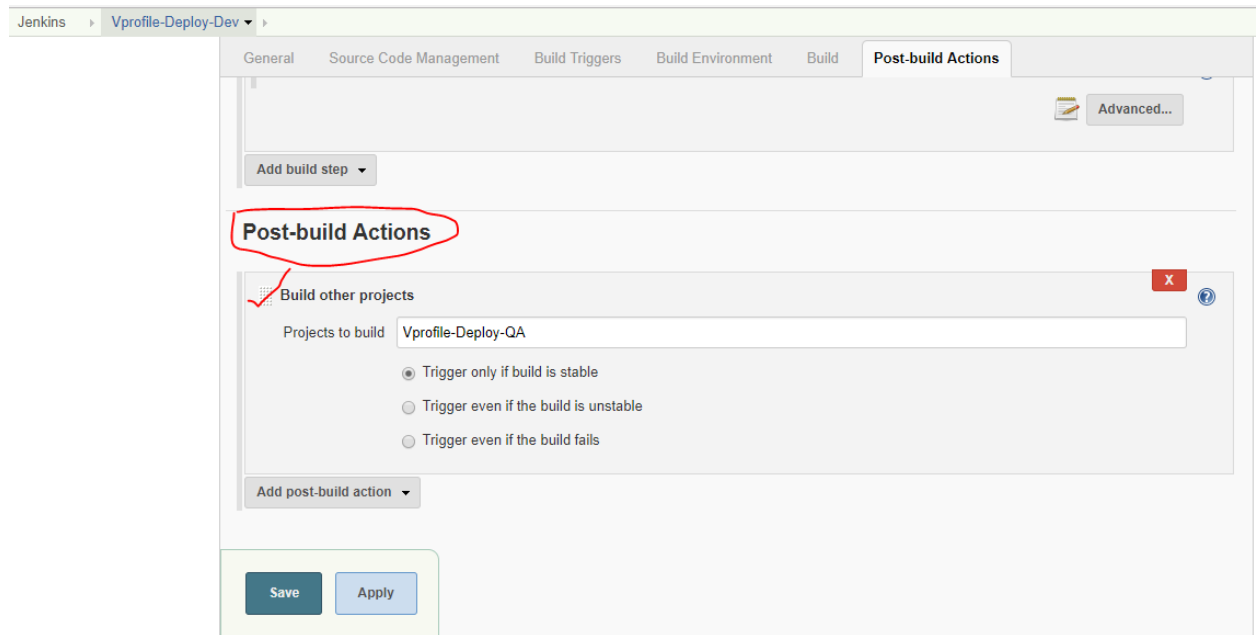
This build requires parameters:

Time: 201805111926
ID: 1
Version: 201805111926-1|

Build

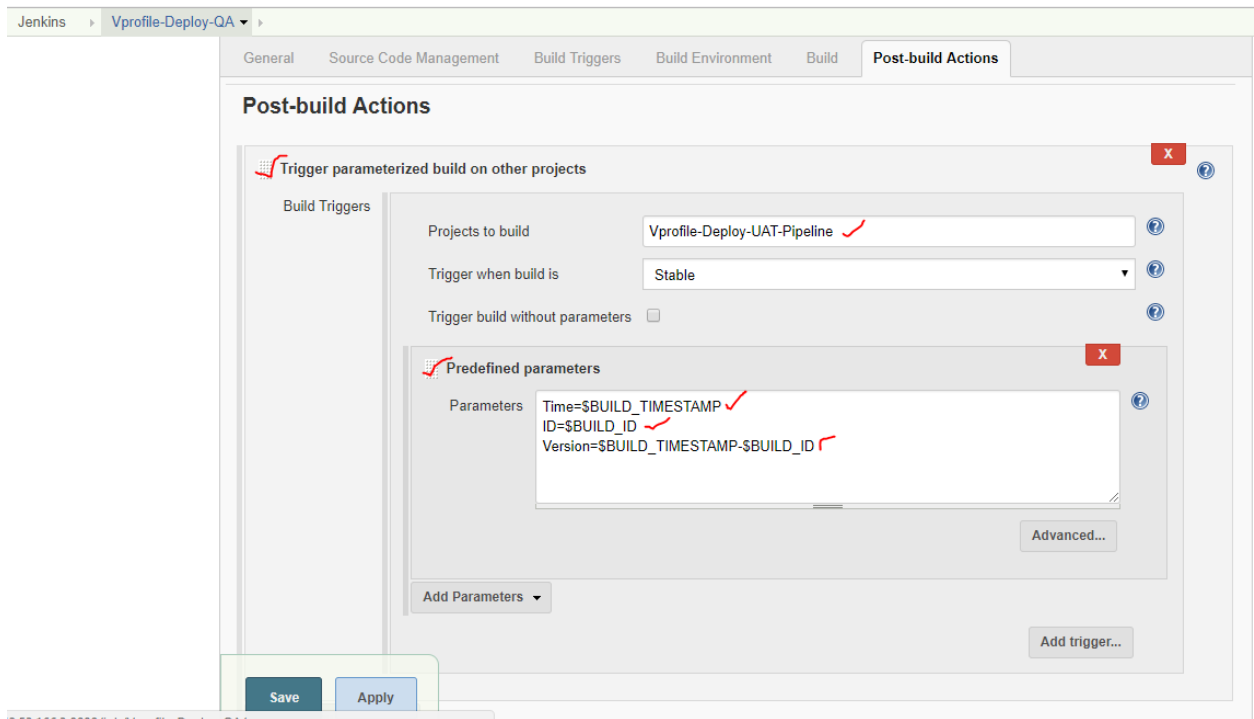
- ❖ And click on Build so build will get run and same artifact from QA as per our input it is going to deploy into UAT Tomcat server.
- ❖ Build should say success and we can verify both the things like above screen (please verify checks we done for first job above). Then it is success.

- ✓ We just finished our all required 3 jobs setup and deployed to all 3 tomcat servers.
- ✓ Now we can make all 3 jobs in pipeline means we can make up stream and down stream projects and these jobs can be run in a sequence manner. For that we need Build pipeline plugin

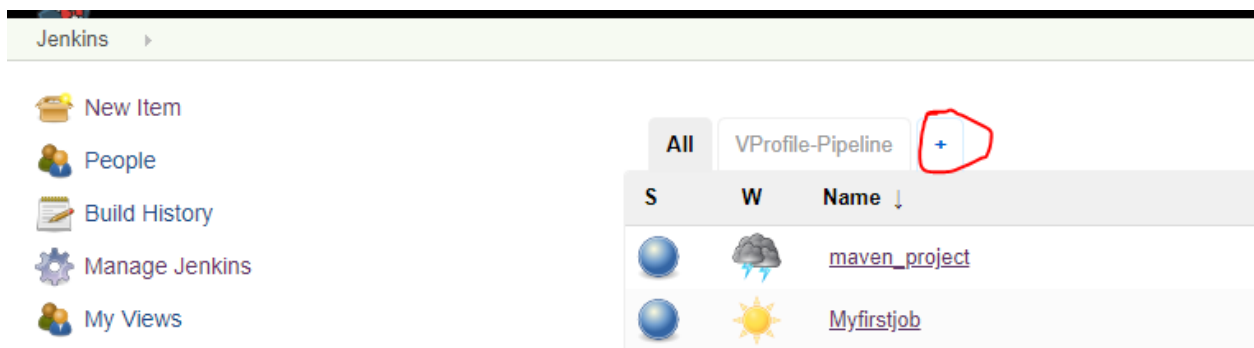


Our UAT job is parameterized means when we want to run this job we have to provide parameters then only this UAT can be run. But Build pipeline we don't get any interface to give user inputs so here we have to give parameters via remotely when this build start automatically from QA job in pipeline.

So first we have to install Trigger parameterized build on other projects and configure as follows on QA job.



- ✓ Once done with upstream and downstream setup we have to create delivery pipeline in a sequence.
- ✓ After installing build plug in just go to Jenkins home page and able to plus symbol



View name

☒ **Build Pipeline View**
Shows the jobs in a build pipeline view. The complete pipeline of jobs that a version propagates through are shown as a row in the view.

☐ **List View**
Shows items in a simple list format. You can choose which jobs are to be displayed in which view.

☐ **My View**
This view automatically displays all the jobs that the current user has an access to.

Name

Description

[Plain text] [Preview](#)

Filter build queue ☐

Filter build executors ☐

Build Pipeline View Title

Pipeline Flow

Layout

Based on upstream/downstream relationship

This layout mode derives the pipeline structure based on the upstream/downstream trigger relationship between jobs. This is the only out-of-the-box supported layout mode, but is open for extension.

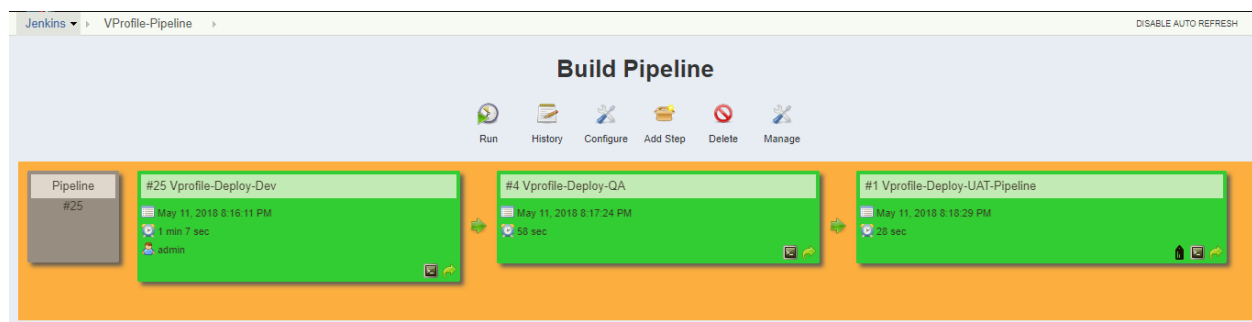
Upstream / downstream config

Select Initial Job

Vprofile-Deploy-Dev

Trigger Options

Save and Run this pipeline.



Need to click on Run symbol so that builds can be deployed to all 3 environments and we can go and check on web url to access our project on 3 servers of tomcat. Thanks Siva