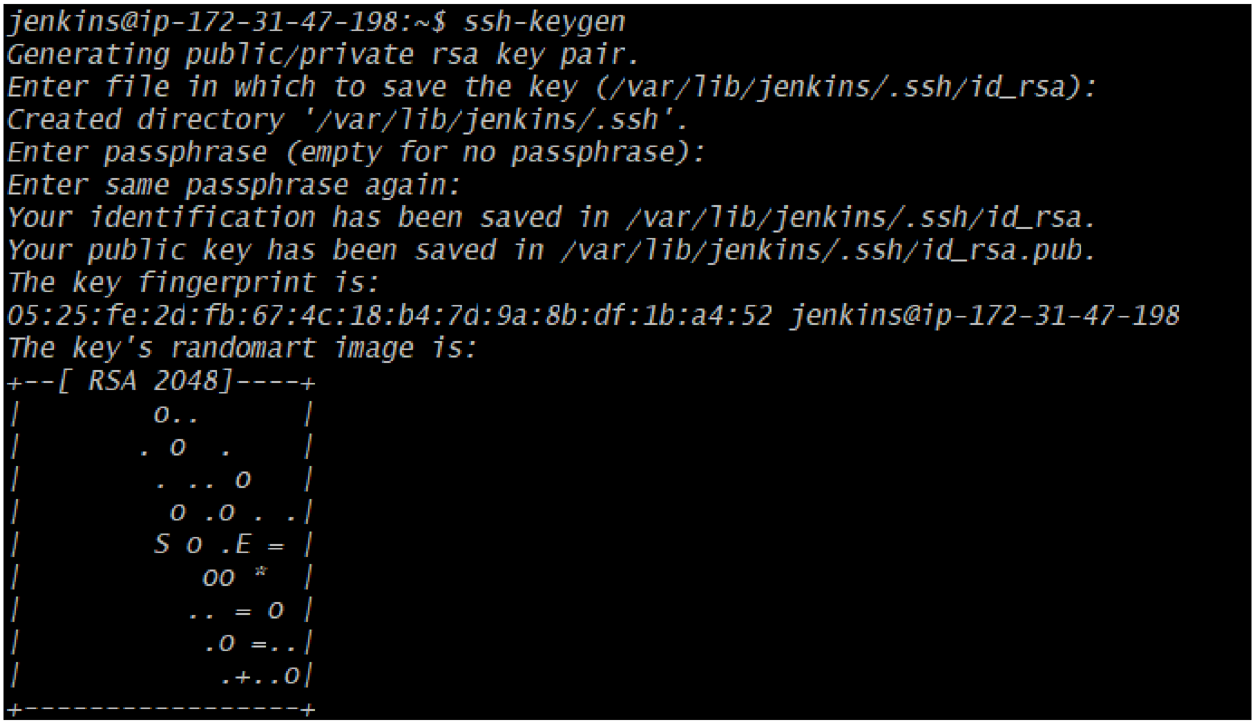
Continuous Delivery Using Jenkins

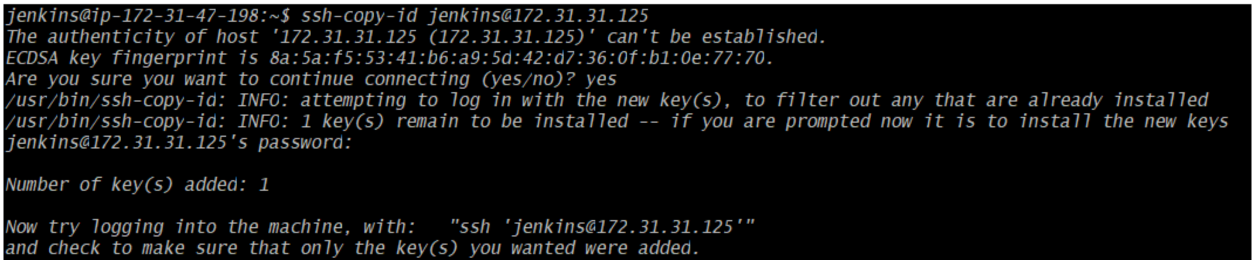
Continuous Delivery in Jenkins can be achieved by using multiple ways. Lets start with Basic approach i.e. using shell script to deploy a war to tomcat

1. Create the key for the Jenkins user
   Login as Jenkins User using command “su Jenkins”
   Create the key using command ssh-keygen



```
jenkins@ip-172-31-47-198:~$ ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/var/lib/jenkins/.ssh/id_rsa):
Created directory '/var/lib/jenkins/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /var/lib/jenkins/.ssh/id_rsa.
Your public key has been saved in /var/lib/jenkins/.ssh/id_rsa.pub.
The key fingerprint is:
05:25:fe:2d:fb:67:4c:18:b4:7d:9a:8b:df:1b:a4:52 jenkins@ip-172-31-47-198
The key's randomart image is:
+--[ RSA 2048]----+
|        o..      |
|       . o  .    |
|        . .. o   |
|         o .o . .|
|        S o .E = |
|           oo *  |
|          .. = o |
|           .o =..|
|            .+..o|
+-----------------+
```

2. Login into the other machine where the tomcat is to be installed.
3. Ensure the user with username Jenkins is present with sudo permissions
4. Copy the SSH key from Jenkins machine to Tomcat machine using ssh-copy-id



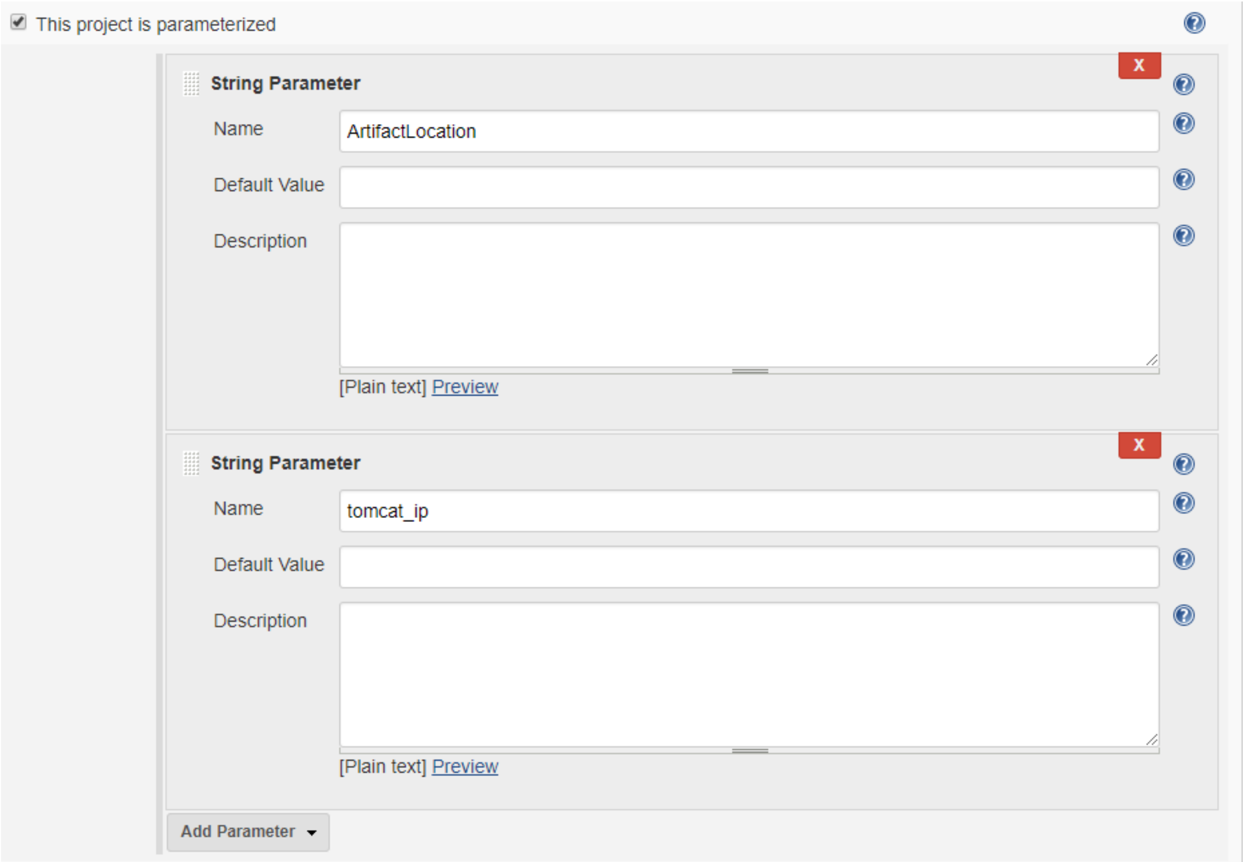
```
jenkins@ip-172-31-47-198:~$ ssh-copy-id jenkins@172.31.31.125
The authenticity of host '172.31.31.125 (172.31.31.125)' can't be established.
ECDSA key fingerprint is 8a:5a:f5:53:41:b6:a9:5d:42:d7:36:0f:b1:0e:77:70.
Are you sure you want to continue connecting (yes/no)? yes
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter out any that are already installed
/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
jenkins@172.31.31.125's password:

Number of key(s) added: 1

Now try logging into the machine, with:   "ssh 'jenkins@172.31.31.125'"
and check to make sure that only the key(s) you wanted were added.
```

5. Test logging into tomcat machine by using ssh jenkins@<ip>
6. Ensure Jenkins user has permission on /var/lib/tomcat8/webapps
7. Create a new Jenkins job with two parameters

Tomcat_ip & ArtifactLocation



☑ This project is parameterized

String Parameter

Name: ArtifactLocation

Default Value:

Description:

[Plain text] Preview

String Parameter

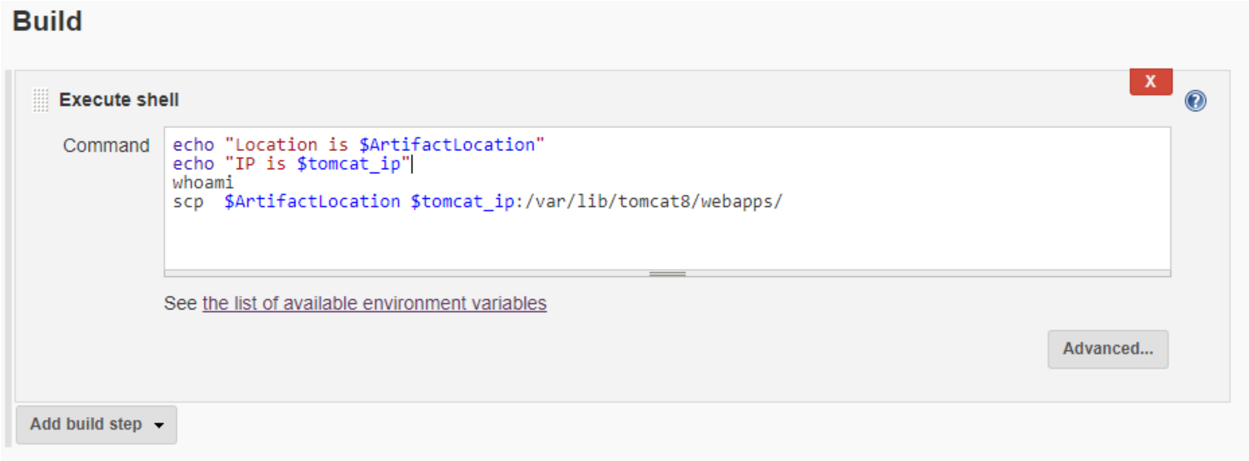
Name: tomcat_ip

Default Value:

Description:

[Plain text] Preview

8. Navigate to Build Step & Use following Script
   “**echo "Location is $ArtifactLocation"**
   echo "IP is $tomcat_ip"
   whoami
   scp $ArtifactLocation $tomcat_ip:/var/lib/tomcat8/webapps/”



Build

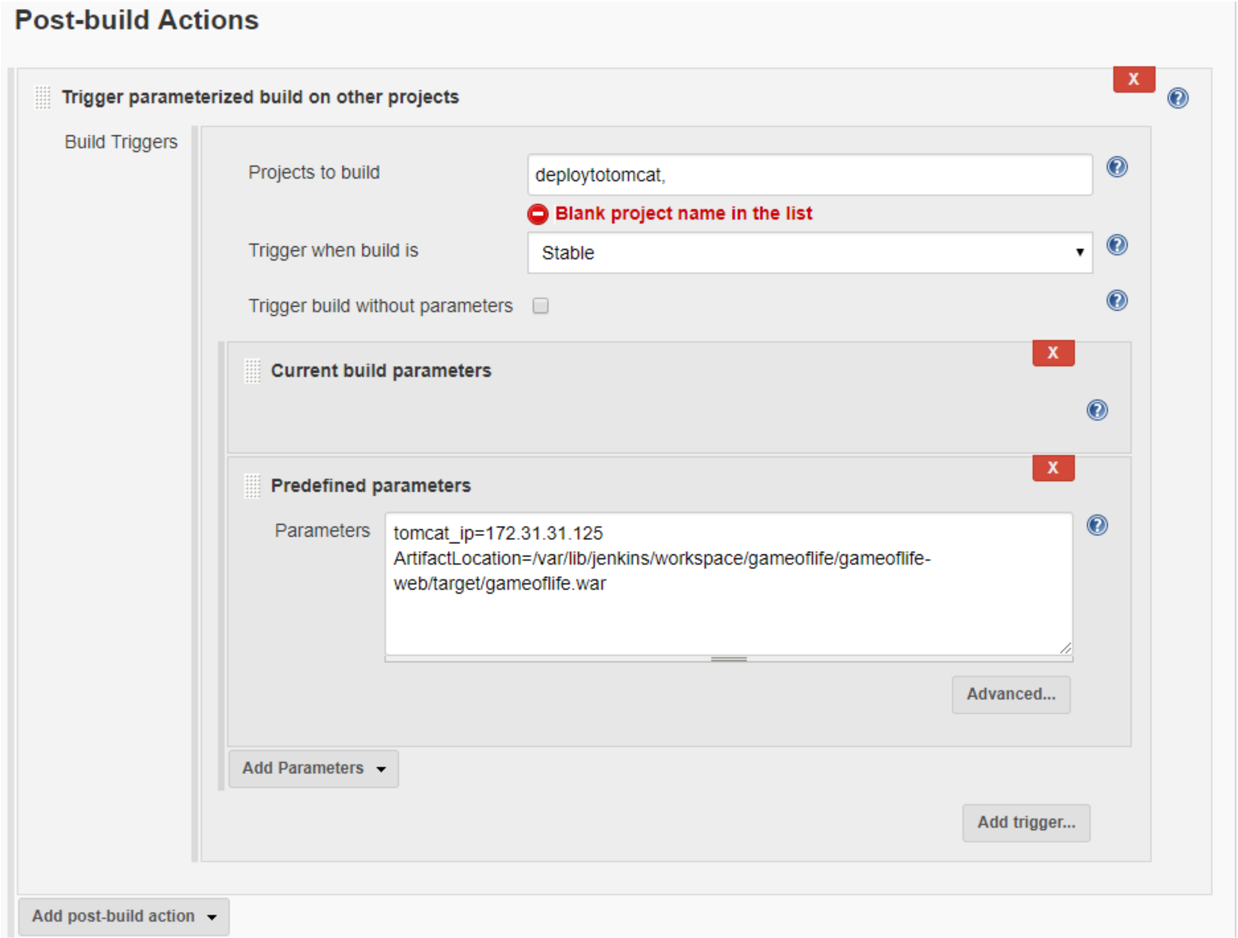
Execute shell

Command

```
echo "Location is $ArtifactLocation"
echo "IP is $tomcat_ip"
whoami
scp  $ArtifactLocation $tomcat_ip:/var/lib/tomcat8/webapps/
```

See the list of available environment variables

9. Installed Parametrized Trigger Plugin
10. Navigate to the project where the artifact is built (game-of-life) in this case and select post-build actions



Post-build Actions

Trigger parameterized build on other projects

Build Triggers

Projects to build: deploytotomcat,

Blank project name in the list

Trigger when build is: Stable

Trigger build without parameters □

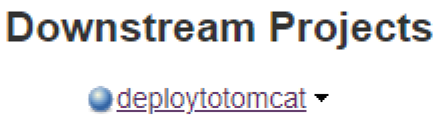
Current build parameters

Predefined parameters

Parameters:

```
tomcat_ip=172.31.31.125
ArtifactLocation=/var/lib/jenkins/workspace/gameoflife/gameoflife-web/target/gameoflife.war
```

11. Save the Job and you should be able to see the following section in the game-of-life job



Downstream Projects

deploytotomcat

12. Build the project & see both the projects being executed one after other.

********************************END**