

## Task Requirements:

```
# Opstree-task
## Create a Jenkins Job that will take two inputs:
1) Instance Id
2) Instance Type

## The Jenkins Job should perform 3 operations:
1) Change the instance type of instance with the provided instance type.
2) Perform an operation using ansible on the provided instance
3) Scale back the instance to the previous instance type.
```

## Here I slightly changed the requirements:

```
# Opstree-task
## Create a Jenkins Job that will take one input:
1) Instance Type

## The Jenkins Job should perform 3 operations:
1) Get the instance id from the tag and change the instance type of instance
with the provided instance type.
2) Perform an operation using ansible on the provided instance
3) Scale back instances to the previous instance type.
```

Created two instances – Jenkins\_server and Testing\_server

Instances (1/2) [Info](#)

Search

|                                     | Name           | Instance ID         | Instance state | Instance type | Status checks         |
|-------------------------------------|----------------|---------------------|----------------|---------------|-----------------------|
| <input checked="" type="checkbox"/> | Jenkins_Server | i-0b73cb693d1533095 | Running        | t2.micro      | 2/2 checks successful |
| <input type="checkbox"/>            | Testing_Server | i-0f244234386bc6da7 | Initializing   | t2.micro      | 1/2 checks successful |

Instance: i-0b73cb693d1533095 (Jenkins\_Server)

**Details** | Security | Networking | Storage | Status checks | Monitoring | Tags

▼ Instance summary [Info](#)

|   |   |
|---|---|
| Instance ID<br>i-0b73cb693d1533095 (Jenkins_Server) | Public IPv4 address<br>54.165.133.53   <a href="#">open address</a> |
|---|---|

Instances (1/2) Info

Search

|                                     | Name           | Instance ID         | Instance state | Instance type | Statu |
|-------------------------------------|----------------|---------------------|----------------|---------------|-------|
| <input type="checkbox"/>            | Jenkins_Server | i-0b73cb693d1533095 | Running        | t2.micro      | 2/    |
| <input checked="" type="checkbox"/> | Testing_Server | i-0f244234386bc6da7 | Running        | t2.micro      | Ini   |

Instance: i-0f244234386bc6da7 (Testing\_Server)

Details Security Networking Storage Status checks Monitoring Tags

▼ Instance summary Info

|                                      |                             |
|--------------------------------------|-----------------------------|
| Instance ID                          | Public IPv4 address         |
| i-0f244234386bc6da7 (Testing_Server) | 50.17.25.145   open address |
| IPv6 address                         | Instance state              |

In the Jenkins\_Server –

1. Jenkins pre-requisites (jdk and other dependencies)
2. Installed Jenkins
3. Ansible from amazon-linux-extras
  - a. Created ansadmin user in both Jenkins\_server and Testing\_Server and created ssh-keys in Jenkins\_server and copied public key and pasted in Testing\_server – ansadmin -> .ssh -> authorized\_keys (give permissions to .ssh folder and authorized\_keys)
  - b. Visudo – added ansadmin entry
  - c. Changed Passwordlessauthentication yes in /etc/ssh/sshd\_config then reloaded the sshd
  - d. Verified ssh connectivity from ansible control node (jenkins\_server) to target node (testing\_server)
  - e. In Jenkins server where ansible is installed under /etc/ansible/hosts – added webserver host group and given testing\_server private ip

```
[webserver]
172.31.18.183
[root@ip-172-31-18-249 ~]#
```

- i.
4. In Jenkins installed Ansible plugin (configured ansible path in global tool configuration and configured ansadmin credentials as global tool configuration), Cloudbees aws plugin (configured aws-keys as username and password), configured git credentials

Below is the pipeline – <https://github.com/madhusudhan7492/Opstree-task-python/blob/master/Jenkinsfile>

**This pipeline code is all about:**

1. Taking the code from the Github repository
2. Verifying the instance which is having tags as Testing\_Server (python script)
3. Verifying the instance type (in this case it is t2.micro)
4. Stopping the instance
5. Change the instance type (configured the parameterized input)

## Pipeline Opstree-python-pipeline

This build requires parameters:

InstanceType

t2.nano

Build

6. Start the instance
7. Run the ansible-playbook (install httpd, start the service and verify the service is running)
8. Stop the instance
9. Change the instance to its original (t2.micro)
10. Start the instance

```
pipeline {
  agent any
  stages {
    stage('Checkout SCM') {
      steps {
        checkout([$class: 'GitSCM', branches: [
          [name: '*/master']
        ], doGenerateSubmoduleConfigurations: false, extensions: [],
        submoduleCfg: [], userRemoteConfigs: [
          [credentialsId: 'Github_creds', url:
'https://github.com/madhusudhan7492/Opstree-task-python.git']
        ]])
      }
    }

    stage('Run Start and Change Instance Type script') {
      steps {

        withCredentials([
          [$class: 'UsernamePasswordMultiBinding', credentialsId: 'aws-creds',
usernameVariable: 'AWS_ACCESS_KEY_ID', passwordVariable:
'AWS_SECRET_ACCESS_KEY']
        ]) {
          sh '/usr/bin/python3 -u startAndChangeInstance.py'
        }

      }
    }

    stage('Run ansible playbook') {
      steps {
        //here main.yml file is in the cloned repository
        ansiblePlaybook credentialsId: 'ansadmin', disableHostKeyChecking:
true, installation: 'ansible', playbook: 'main.yml'
      }
    }
  }
}
```

```

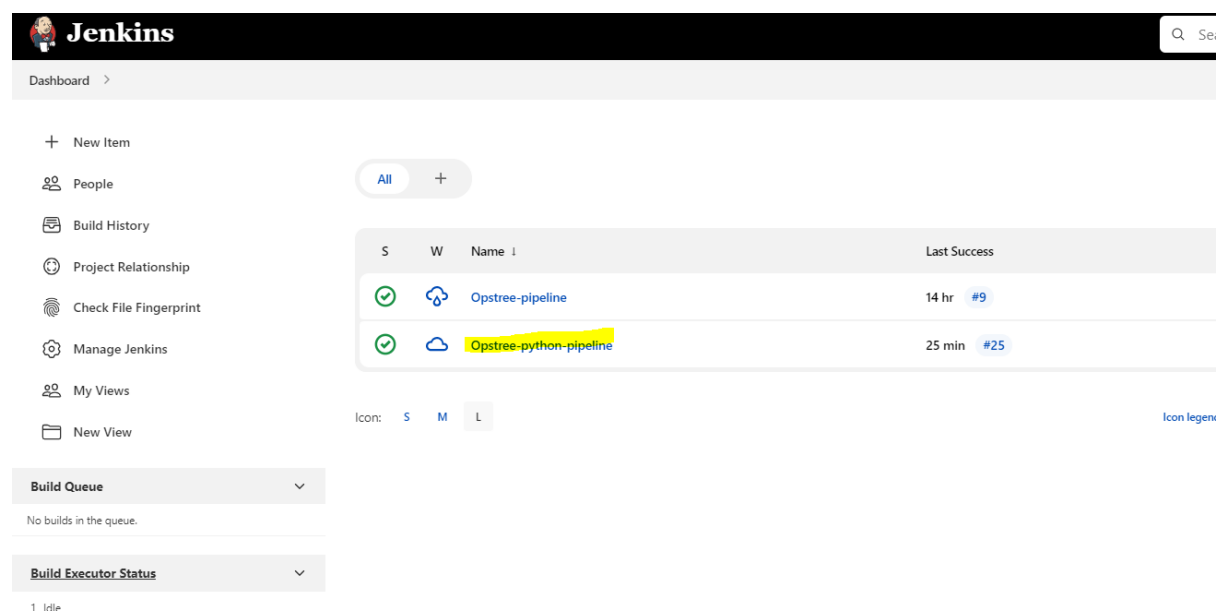
stage('Run flipback and start instance script') {
    steps {

        withCredentials([
            [$class: 'UsernamePasswordMultiBinding', credentialsId: 'aws-creds',
            usernameVariable: 'AWS_ACCESS_KEY_ID', passwordVariable:
            'AWS_SECRET_ACCESS_KEY']
        ]) {
            sh '/usr/bin/python3 -u flipbackAndStartInstance.py'
        }
    }
}
}
}

```

## Demo:

### Created an Opstree-python-pipeline



The screenshot shows the Jenkins dashboard with the following components:

- Header:** Jenkins logo and a search bar.
- Left Sidebar:**
  - + New Item
  - People
  - Build History
  - Project Relationship
  - Check File Fingerprint
  - Manage Jenkins
  - My Views
  - New View
- Main Content Area:**
  - Build Queue:** No builds in the queue.
  - Build Executor Status:** 1 Idle
  - Build History Table:**

| S | W | Name                    | Last Success |
|---|---|-------------------------|--------------|
| ✓ | ☁ | Opstree-pipeline        | 14 hr #9     |
| ✓ | ☁ | Opstree-python-pipeline | 25 min #25   |

### Build with parameters

#### Pipeline Opstree-python-pipeline

This build requires parameters:

InstanceType

t2.nano

Build

### After running the job

|  |                             | Declarative:<br>Checkout SCM | Checkout SCM | Run Start and<br>Change<br>Instance Type<br>script | Run ansible<br>playbook | Run flipback<br>and start<br>instance script |
|--|-----------------------------|------------------------------|--------------|--|-------------------------|--|
| Average stage times:<br>(Average <u>full</u> run time: ~3min 1s) |                             | 221ms                        | 194ms        | 1min 26s   | 7s                      | 1min 3s                                      |
| #25  | Jul 03<br>12:41<br>1 commit | 229ms                        | 198ms        | 1min 26s   | 7s                      | 1min 26s                                     |



BUILD\_OUTPUT.txt

### **Build output:**

### **Verifying the output – Httpd is installed and running successfully**

Test Page

This page is used to test the proper operation of the Apache HTTP server after it has been installed. If you can read this page, it means that the Apache HTTP server installed at this site is working properly.

**If you are a member of the general public:**

The fact that you are seeing this page indicates that the website you just visited is either experiencing problems, or is undergoing routine maintenance.

If you would like to let the administrators of this website know that you've seen this page instead of the page you expected, you should send them e-mail. In general, mail sent to the name "webmaster" and directed to the website's domain should reach the appropriate person.

For example, if you experienced problems while visiting `www.example.com`, you should send e-mail to "webmaster@example.com".

**If you are the website administrator:**

You may now add content to the directory `/var/www/html/`. Note that until you do so, people visiting the site will see this page and not your content. To prevent this page from ever being used, follow the instructions in the file `README`.

You are free to use the image below on web sites powered by the Apache HTTP Server: