



ANSIBLE



TASK 3: DESCRIPTION

Deploy a Load Balancer and Multiple Web Servers on AWS instances through ANSIBLE!!

- ☀ Provision EC2 instances through ansible.
- ☀ Retrieve the IP Address of instances using the dynamic inventory concept.
- ☀ Configure the web servers through the ansible role.
- ☀ Configure the load balancer through the ansible role.
- ☀ The target nodes of the load balancer should auto-update as per the status of web servers.

Things to Remember:-

☀ A web server is a computer that runs websites. It's a computer program that distributes web pages as they are requisitioned. The basic objective of the web server is to store, process and deliver web pages to the users. This intercommunication is done using Hypertext Transfer Protocol (HTTP).

☀ Load Balancer performs the following functions: Distributes client requests or network load efficiently across multiple servers. Ensures high availability and reliability by sending requests only to servers that are online. Provides the flexibility to add or subtract servers as demand dictates.

☀ Ansible is an open-source IT Configuration Management, Deployment & Orchestration tool. It aims to provide large productivity gains to a wide variety of automation challenges. This tool is very simple to use yet powerful enough to automate complex multi-tier IT application environments.



+



Step 1 - It is important to install boto and boto3 before launching the instance.

```
[root@localhost ~]# pip3 list | grep boto
DEPRECATION: The default format will switch to columns in the future. You can use --format=(legacy|columns) (or define a format=(legacy|columns) in your pip.conf under the [list] section) to disable this warning.
boto (2.49.0)
boto3 (1.14.43)
botocore (1.17.43)
```

Step 2 - Creation of a role for launching the instance on top of AWS.

Commands:-

```
mkdir /etc/myroles
```

```
mkdir /etc/myroles
```

```
cd /etc/myroles
```

```
ansible-galaxy init ec2_role
```

```
[root@CN myroles]# cd ec2_role
[root@CN ec2_role]# ls
defaults  files  handlers  meta  README.md  tasks  templates  tests  vars
[root@CN ec2_role]# vim tasks/main.yml
```

```
---
# tasks file for ec2_role
- name: Provisioning Of EC2 On AWS
  ec2:
    aws_access_key: "{{ myuser }}"
    aws_secret_key: "{{ mypass }}"
    key_name: AWS_KEY_PAIR
    instance_type: t2.micro
    image: ami-0ebc1ac48dfd14136
    assign_public_ip: yes
    region: ap-south-1
    count: 3
    group_id: sg-044803349c65918aa
    vpc_subnet_id: subnet-00405ed5791820019
    instance_tags:
      Name: Web

- name: Add Host for further Contact
  add_host:
    name: "Web"
    groups: "Webservers"
```

```

- name: Provisioning Of LB EC2 On AWS
  ec2:
    aws_access_key: "{{ myuser }}"
    aws_secret_key: "{{ mypass }}"
    key_name: AWS_KEY_PAIR
    instance_type: t2.micro
    image: ami-0eabc1ac48dfd14136
    assign_public_ip: yes
    region: ap-south-1
    count: 1
    group_id: sg-044803349c65918aa
    vpc_subnet_id: subnet-00405ed5791820019
    instance_tags:
      Name: LoadBalancer

- name: Add Host for Further Contact
  add_host:
    name: "Load Balancer"
    groups: "LB"

```

Step 3:- Inside the vars/main.yml file, we have to pass the AWS Access Key and Secret Key. We have to provide Access key inside "myuser" variable and Secret key inside "mypass" variable

Step 4:- Inside Ansible Configuration file.

`vim /etc/ansible/ansible.cfg`

```

[defaults]
inventory=/etc/myhosts.txt
host_key_checking=False
roles_path=/etc/myroles
private_key_file=/root/AWS_KEY_PAIR.pem
remote_user=ec2-user
ask_pass=False

[privilege_escalation]
become=True
become_user=root
become_ask_pass=False

```

`cd`

`cd /projects`

`vim launch.yml`

```
roles:
  - role: ec2_role
```

Step 5:- Now run Ansible playbook code using ansible-playbook launch.yml cmd

```
root@CN ~]#
root@CN ~]# cd /projects
root@CN projects]# ansible-playbook launch.yml
WARNING: provided hosts list is empty, only localhost is available. Note that the implicit
localhost does not match 'all'

LAY [localhost] *****

ASK [Gathering Facts] *****
ok: [localhost]

ASK [ec2_role : Provisioning Of EC2 On AWS] *****
changed: [localhost]

ASK [ec2_role : Add Host for further Contact] *****
changed: [localhost]

ASK [ec2_role : Provisioning Of LB EC2 On AWS] *****
changed: [localhost]

ASK [ec2_role : Add Host for Further Contact] *****
changed: [localhost]

LAY RECAP *****
localhost : ok=5 changed=4 unreachable=0 failed=0 skipped=0 rescued
ignored=0
```

Screenshot of the Output :-

Welcome to the new instances experience! We're redesigning the EC2 console to make it easier to use. To switch between the old console and the new console, use the New EC2 Experience toggle above the navigation panel. We'll release updates continuously based on customer feedback.

Instances (4) Info

Filter instances

	Name	Instance ID	Instance state	Instance type	Status check	Alarm Status	Availability zone
<input type="checkbox"/>	Web	i-0d85ece066b80a750	Running	t2.micro	2/2 checks ...	No alarms +	ap-south-1a
<input type="checkbox"/>	Web	i-08d3774646a3790b5	Running	t2.micro	Initializing	No alarms +	ap-south-1a
<input type="checkbox"/>	Web	i-01c5047a7d6eb6f61	Running	t2.micro	Initializing	No alarms +	ap-south-1a
<input type="checkbox"/>	LoadBalancer	i-04438dc9f763a1fc6	Running	t2.micro	Initializing	No alarms +	ap-south-1a

Instances (4/4) Info

Filter instances

	Name	Instance ID	Instance state	Instance type	Status check	Alarm Status	Availability zone
<input checked="" type="checkbox"/>	Web	i-0d85ece066b80a750	Running	t2.micro	2/2 checks ...	No alarms +	ap-south-1a
<input checked="" type="checkbox"/>	Web	i-08d3774646a3790b5	Running	t2.micro	2/2 checks ...	No alarms +	ap-south-1a
<input checked="" type="checkbox"/>	Web	i-01c5047a7d6eb6f61	Running	t2.micro	2/2 checks ...	No alarms +	ap-south-1a
<input checked="" type="checkbox"/>	LoadBalancer	i-04438dc9f763a1fc6	Running	t2.micro	2/2 checks ...	No alarms +	ap-south-1a

Instances: i-04438dc9f763a1fc6 (LoadBalancer), i-01c5047a7d6eb6f61 (Web), i-08d3774646a3790b5 (Web), i-0d85ece066b80a750 (Web)

Monitoring

Add to dashboard 1h 3h 12h 1d 3d 1w custom

CPU utilization (%) Various units

Status check failed (any) (count) 1 No data available. 0.5 Try adjusting the dashboard time range.

Status check failed (instanc... 1 No data available. 0.5 Try adjusting the dashboard time range.

Status check failed (system) (c... 1 No data available. 0.5 Try adjusting the dashboard time range.

Step6- To get the Instance IP dynamically in the inventory, we have to download ec2.py and ec2.init files

mkdir /mydb

cd /mydb

yum install wget

wget https://raw.githubusercontent.com/ansible/ansible/stable-2.9/contrib/inventory/ec2.py


```
[root@CN mydb]# wget https://raw.githubusercontent.com/ansible/ansible/stable-2.9/contrib/inventory/ec2.py
--2020-08-23 08:16:14-- https://raw.githubusercontent.com/ansible/ansible/stable-2.9/contrib/inventory/ec2.py
Resolving raw.githubusercontent.com (raw.githubusercontent.com)... 199.232.20.133
Connecting to raw.githubusercontent.com (raw.githubusercontent.com)|199.232.20.133|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 73130 (71K) [text/plain]
Saving to: 'ec2.py'

ec2.py          100%[=====>]  71.42K  --.-KB/s    in 0.09s

2020-08-23 08:16:16 (798 KB/s) - 'ec2.py' saved [73130/73130]
```

```
[root@CN mydb]# ls
ec2.py
[root@CN mydb]# chmod +x ec2.py
[root@CN mydb]# ls
ec2.py
[root@CN mydb]# vim ec2.py
```

```
#!/usr/bin/python3

'''
EC2 external inventory script
=====

Generates inventory that Ansible can understand by making API request to
AWS EC2 using the Boto library.

NOTE: This script assumes Ansible is being executed where the environment
variables needed for Boto have already been set:
    export AWS_ACCESS_KEY_ID='AK123'
    export AWS_SECRET_ACCESS_KEY='abc123'

Optional region environment variable if region is 'auto'

This script also assumes that there is an ec2.ini file alongside it. To specify a
different path to ec2.ini, define the EC2_INI_PATH environment variable:

    export EC2_INI_PATH=/path/to/my_ec2.ini

If you're using eucalyptus you need to set the above variables and
you need to define:

    export EC2_URL=http://hostname_of_your_cc:port/services/Eucalyptus

If you're using boto profiles (requires boto>=2.24.0) you can choose a profile
using the --boto-profile command line argument (e.g. ec2.py --boto-profile prod) or using
the AWS_PROFILE variable:

    AWS_PROFILE=prod ansible-playbook -i ec2.py myplaybook.yml

For more details, see: http://docs.pythonboto.org/en/latest/boto\_config\_tut.html

You can filter for specific EC2 instances by creating an environment variable
named EC2_INSTANCE_FILTERS, which has the same format as the instance_filters
-- INSERT --
```

```

[root@CN mydb]#
[root@CN mydb]# wget https://raw.githubusercontent.com/ansible/ansible/stable-2.9/contrib/inventor
ec2.ini
--2020-08-23 07:10:34-- https://raw.githubusercontent.com/ansible/ansible/stable-2.9/contrib/inve
ory/ec2.ini
Resolving raw.githubusercontent.com (raw.githubusercontent.com)... 199.232.20.133
Connecting to raw.githubusercontent.com (raw.githubusercontent.com)|199.232.20.133|:443... connect
.
HTTP request sent, awaiting response... 200 OK
Length: 9529 (9.3K) [text/plain]
Saving to: 'ec2.ini'

ec2.ini                100%[=====>]    9.31K  --.-KB/s    in 0s

2020-08-23 07:10:35 (38.3 MB/s) - 'ec2.ini' saved [9529/9529]

```

Step7- Now inside the ansible configuration file, change the inventory to /mydb

```

[defaults]
inventory=/mydb_
host_key_checking=False
roles_path=/etc/myroles
private_key_file=/root/AWS_KEY_PAIR.pem
remote_user=ec2-user
ask_pass=False

[privilege_escalation]
become=True
become_user=root
become_ask_pass=False

```

Step8- Use command **ansible all --list-hosts** which will show all the host IP's.

```

root@CN mydb]# ansible all --list-hosts
[WARNING]: Invalid characters were found in group names but not replaced, use -vvvv to see details
hosts (4):

```

```

35.243.207.8
34.75.100.42
35.227.82.54
34.75.132.197

```


Step9 - First we have to create roles for configuring the webserver and haproxy software inside the EC2 instances.

```
cd /etc/myroles
```

```
ansible-galaxy init Webserver
```

```
ansible-galaxy init lbserver
```

```
[root@CN ~]# cd /etc/myroles
[root@CN myroles]# cd Webserver
[root@CN Webserver]# ls
defaults  files  handlers  meta  README.md  tasks  templates  tests  vars
[root@CN Webserver]# vim tasks/main.yml_
```

```
# tasks file for Webserver
- name: "Install httpd"
  package:
    name: "httpd"
    state: present

- name: "Copy Content"
  copy:
    content: "Hello, This is {{ansible_hostname}}"
    dest: "/var/www/html/index.html"

- name: "Start httpd Service"
  service:
    name: "httpd"
    state: started
```

For Loadbalancer server:-

```
[root@CN Webserver]# cd /etc/myroles
[root@CN myroles]# cd lbserver
[root@CN lbserver]# ls
'!' defaults files handlers meta README.md tasks templates tests vars
```

```
[root@localhost ~]# ansible-galaxy list
# /root/ansible/myroles
- webserver, (unknown version)
- mydb, (unknown version)
- lbserver, (unknown version)
- targetserver, (unknown version)
```

Make changes in Haproxy.cfg file using `vim haproxy.cfg` cmd

Bind *:8080

```
acl url_static      path_beg      -i /static /images /javascript /

acl url_static      path_end      -i .jpg .gif .png .css .js

use_backend static   if url_static
default_backend      app

-----
tatic backend for serving up images, stylesheets and such
-----

kend static
  balance      roundrobin
  server       static 127.0.0.1:4331 check

-----
ound robin balancing between the various backends
-----

kend app
  balance      roundrobin
  {% for host in groups['web'] %}
  server app1 {{ host }}:80 check
  {% endfor %}
```

Use this cmd to open `vim tasks/main.yml` file.

```
# tasks file for lbserver
- name: "Install Haproxy Software"
  package:
    name: "haproxy"
    state: present

- name: "Copy Haproxy Config File"
  template:
    src: "haproxy.cfg"
    dest: "/etc/haproxy/haproxy.cfg"
  notify: LB Restart

- name: "Start Haproxy Service"
  service:
    name: "haproxy"
    state: started
```

Use this cmd to open `vim handlers/main.yml` file

```
# handlers file for lbserver
- name: LB Restart
  service:
    name: "haproxy"
    state: restarted
```

Step10- Now we have to create a file inside the projects folder for the webserver and lbserver roles .

```
hosts: Webservers
roles:
  - role: Webserver

hosts: LB
roles:
  - role: lbserver
```

Step11- Make some changes inside the inventory file and the ansible.cfg file.

`vim /etc/myhosts.txt`

```
[Webservers]
35.243.207.8
34.75.100.42
35.227.82.54

[LB]
34.75.132.197
```

`vim /etc/ansible/ansible.cfg`

```
[defaults]
inventory=/etc/myhosts.txt
host_key_checking=False
roles_path=/etc/myroles
private_key_file=/root/AWS_KEY_PAIR.pem
remote_user=ec2-user
ask_pass=False

[privilege_escalation]
become=True
become_user=root
become_ask_pass=False
```


Step12- Now, for running the setup.yml playbook we have to use **ansible-playbook setup.yml** command.

```
[root@CN Webserver]# cd /projects
[root@CN projects]# ansible-playbook setup.yml

PLAY [Webservers] *****

TASK [Gathering Facts] *****
[WARNING]: Platform linux on host 35.243.207.8 is using the discovered Python interpreter at
/usr/bin/python, but future installation of another Python interpreter could change this. See
https://docs.ansible.com/ansible/2.9/reference\_appendices/interpreter\_discovery.html for more
information.
ok: [35.243.207.8]
[WARNING]: Platform linux on host 34.75.100.42 is using the discovered Python interpreter at
/usr/bin/python, but future installation of another Python interpreter could change this. See
https://docs.ansible.com/ansible/2.9/reference\_appendices/interpreter\_discovery.html for more
information.
ok: [34.75.100.42]
[WARNING]: Platform linux on host 35.227.82.54 is using the discovered Python interpreter at
/usr/bin/python, but future installation of another Python interpreter could change this. See
https://docs.ansible.com/ansible/2.9/reference\_appendices/interpreter\_discovery.html for more
information.
ok: [35.227.82.54]
```

```
TASK [Webserver : Install httpd] *****
changed: [35.243.207.8]
changed: [34.75.100.42]
changed: [35.227.82.54]

TASK [Webserver : Copy Content] *****
changed: [35.243.207.8]
changed: [34.75.100.42]
changed: [35.227.82.54]

TASK [Webserver : Start httpd Service] *****
changed: [35.243.207.8]
changed: [34.75.100.42]
changed: [35.227.82.54]
```



```
PLAY [LB] *****
TASK [Gathering Facts] *****
[WARNING]: Platform linux on host 34.75.132.197 is using the discovered Python interpreter at
/usr/bin/python, but future installation of another Python interpreter could change this. See
https://docs.ansible.com/ansible/2.9/reference\_appendices/interpreter\_discovery.html for more
information.
ok: [34.75.132.197]

TASK [lbserver : Install Haproxy Software] *****
changed: [34.75.132.197]

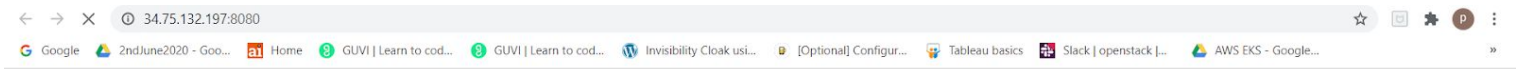
TASK [lbserver : Copy Haproxy Config File] *****
changed: [34.75.132.197]

TASK [lbserver : Start Haproxy Service] *****
changed: [34.75.132.197]

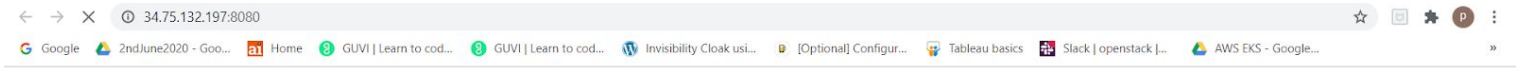
RUNNING HANDLER [lbserver : LB Restart] *****
changed: [34.75.132.197]

PLAY RECAP *****
34.75.132.197      : ok=5    changed=4    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
35.243.207.8      : ok=4    changed=2    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
34.75.100.42      : ok=4    changed=2    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
35.227.82.54      : ok=4    changed=2    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```

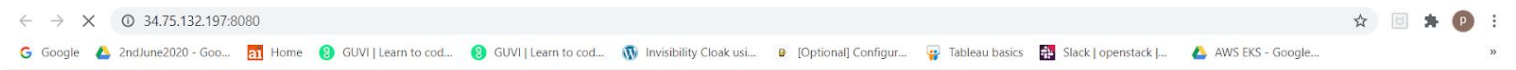
Main Output:-



Hello, This is ip-172-31-43-23



Hello, This is ip-172-31-35-43



Hello, This is ip-172-31-34-147

aws

Services

Pratik

Mumbai

Support

New EC2 Experience

Tell us what you think

EC2 Dashboard

Events

Tags

Limits

Instances

Instance Types

Launch Templates

Spot Requests

Savings Plans

Reserved Instances

Dedicated Hosts

Capacity Reservations

Images

AMIs

Elastic Block Store

Volumes

Snapshots

Lifecycle Manager

Successfully terminated i-0d85ece066b80a750

Successfully terminated i-08d3774646a3790b5

Successfully terminated i-01c5047a7d6eb6f61

Successfully terminated i-04438dc9f763a1fc6

Instances (1/4)

Info

Connect

Instance state

Actions

Launch instances

Filter instances

	Name	Instance ID	Instance state	Instance type	Status check	Alarm Status	Availability zone
<input type="checkbox"/>	Web	i-0d85ece066b80a750	Shutting-down	t2.micro	2/2 checks ...	No alarms +	ap-south-1a
<input type="checkbox"/>	Web	i-08d3774646a3790b5	Shutting-down	t2.micro	2/2 checks ...	No alarms +	ap-south-1a
<input type="checkbox"/>	Web	i-01c5047a7d6eb6f61	Shutting-down	t2.micro	2/2 checks ...	No alarms +	ap-south-1a
<input checked="" type="checkbox"/>	LoadBalancer	i-04438dc9f763a1fc6	Shutting-down	t2.micro	2/2 checks ...	No alarms +	ap-south-1a

Instance: i-04438dc9f763a1fc6 (LoadBalancer)

Details

Security

Networking

Storage

Status Checks

Monitoring

Tags

Instance summary

Info

Instance ID

Public IPv4 address

Private IPv4 addresses

Feedback

English (US)

© 2008 - 2020, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved.

Privacy Policy

Terms of Use

Thanks For Reading!!!