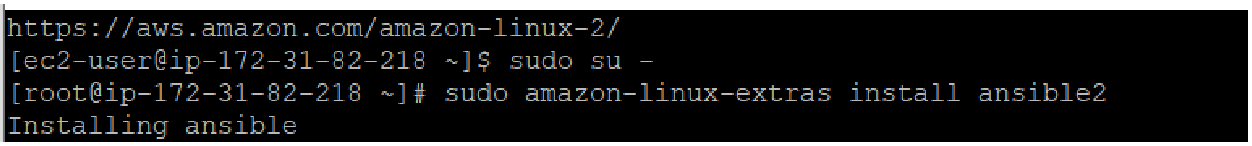
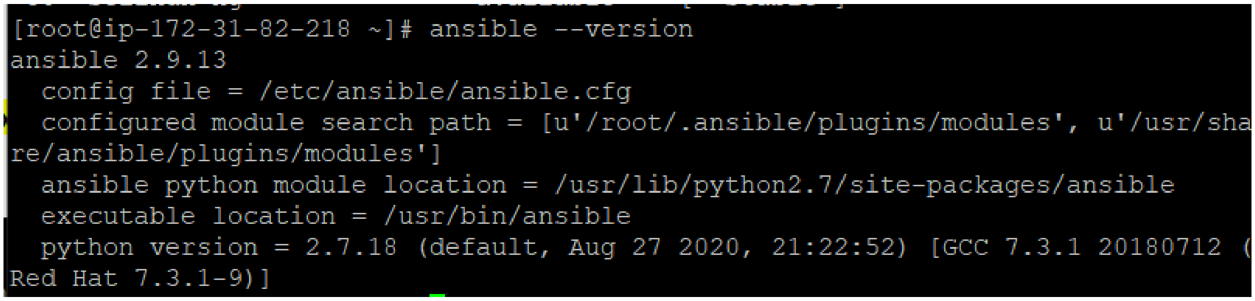
Setting up load balancer with the help of Haproxy and deploying it over AWS

1. Install Ansible in AWS in the root user using **sudo amazon-linux-extras install ansible2**



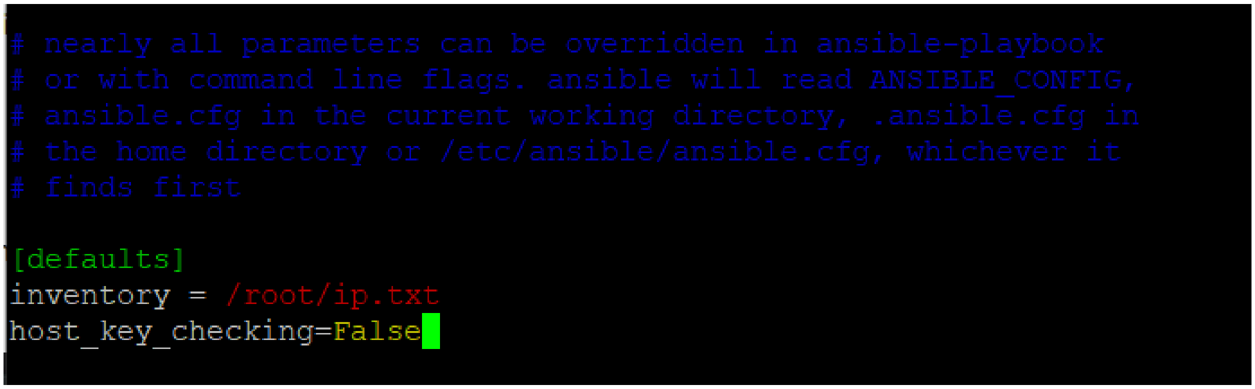
```
https://aws.amazon.com/amazon-linux-2/
[ec2-user@ip-172-31-82-218 ~]$ sudo su -
[root@ip-172-31-82-218 ~]# sudo amazon-linux-extras install ansible2
Installing ansible
```

2. Check if Ansible is successfully installed using **ansible --version**



```
[root@ip-172-31-82-218 ~]# ansible --version
ansible 2.9.13
  config file = /etc/ansible/ansible.cfg
  configured module search path = [u'/root/.ansible/plugins/modules', u'/usr/sha
re/ansible/plugins/modules']
  ansible python module location = /usr/lib/python2.7/site-packages/ansible
  executable location = /usr/bin/ansible
  python version = 2.7.18 (default, Aug 27 2020, 21:22:52) [GCC 7.3.1 20180712 (
Red Hat 7.3.1-9)]
```

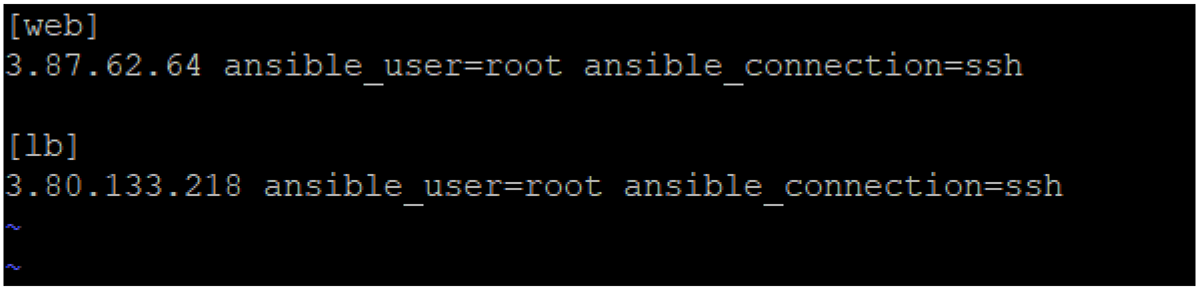
3. Change the config file as shown using **vim /etc/ansible/ansible.cfg**



```
# nearly all parameters can be overridden in ansible-playbook
# or with command line flags. ansible will read ANSIBLE_CONFIG,
# ansible.cfg in the current working directory, .ansible.cfg in
# the home directory or /etc/ansible/ansible.cfg, whichever it
# finds first

[defaults]
inventory = /root/ip.txt
host_key_checking=False
```

Creating an Inventory in AWS:

Make ip.txt file and set one IP as the webserver and the other as the load balancer.



```
[web]
3.87.62.64 ansible_user=root ansible_connection=ssh

[lb]
3.80.133.218 ansible_user=root ansible_connection=ssh
~
~
```

In the ip.txt file, we have not given any password or key for authentication. To provide the key of target node in AWS, the process is as follows:

cd .ssh

ssh-keygen

Press enter key unless you see that box-like figure appearing.

Use ls command to see the files present in the directory.

Open id_rsa.pub file using **vim id_rsa.pub** and copy the content as it is.

Now, go to your target node and follow the steps are stated:

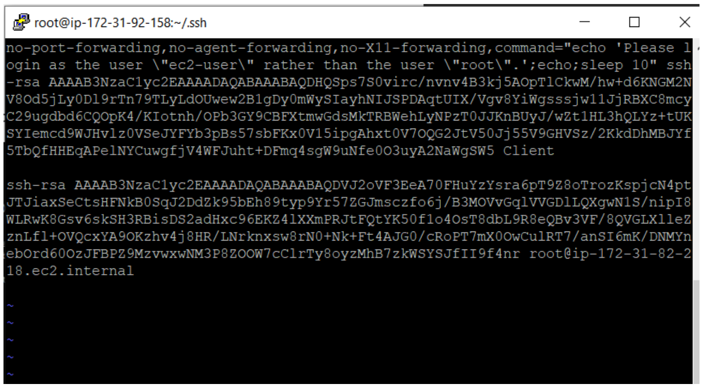
Go to the root directory.

cd .ssh

ssh-keygen

vim authorized_keys

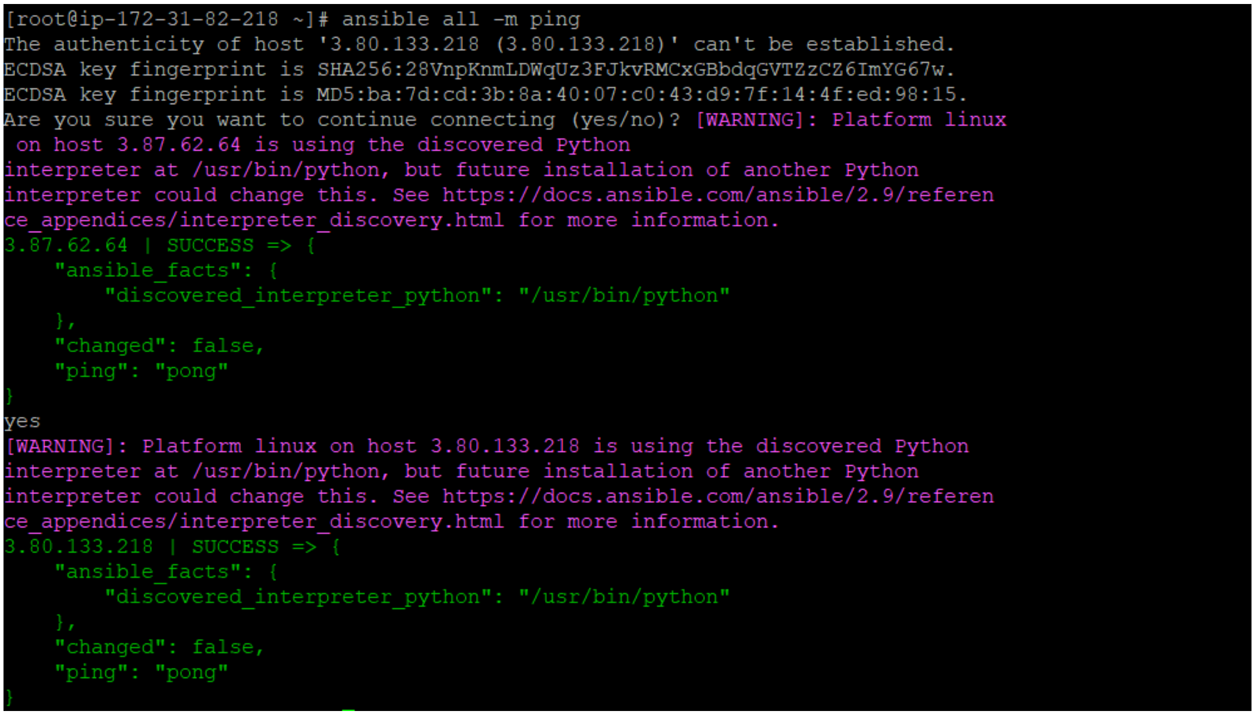
Paste the content that you have copied from the control node below the already written text.



root@ip-172-31-92-158:~/.ssh

```
no-port-forwarding,no-agent-forwarding,no-X11-forwarding,command="echo 'Please l
ogin as the user \"ec2-user\" rather than the user \"root\".';echo;sleep 10" ssh
-rsa AAAAB3NzaC1yc2EAAAADAQABAAABAQDHQSps7S0virc/nvnv4B3kj5AOpTlCkwM/hw+d6KNGM2N
V8Od5jLy0Dl9rTn79TLyLdOUwew2B1gDy0mWySIayhNIJSPDAqtUIX/Vgv8YiWgsssjw11JjRBXC8mcy
C29ugdbd6CQOpK4/KIotnh/OPb3GY9CBFXtmwGdsMkTRBWehLyNPzT0JJKnBUyJ/wZt1HL3hQLYz+tUK
SYIemcd9WJHvlz0VSeJYFYb3pBs57sbFKx0V15ipgAhxt0V7OQG2JtV50Jj55V9GHVSz/2KkdDhMBJYf
5TbQfHHEqAPelNYCuwgfjV4WFJuht+DFmq4sgW9uNfe0O3uyA2NaWgSW5 Client

ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABAQDVJ2oVF3EeA70FHuYzYsra6pT9Z8oTrozKspjcN4pt
JTJiaxSeCtsHFNkB0SqJ2DdZk95bEh89typ9Yr57ZGJmsczfo6j/B3MOVvGqlVVGDlLQXgwN1S/nipI8
WLRwK8Gsv6skSH3RBisDS2adHxc96EKZ4lXXmPRJtFQtYK50f1o4OsT8dbL9R8eQBv3VF/8QVGLXlleZ
znLfl+OVQcxYA9OKzhv4j8HR/LNrknxsw8rN0+Nk+Ft4AJG0/cRoPT7mX0OwCulRT7/anSI6mK/DNMYn
ebOrd60OzJFBPZ9MzvwxwNM3P8ZOOW7cClrTy8oyzMhB7zkWSYSJfII9f4nr root@ip-172-31-82-2
18.ec2.internal

~
~
~
~
~
```

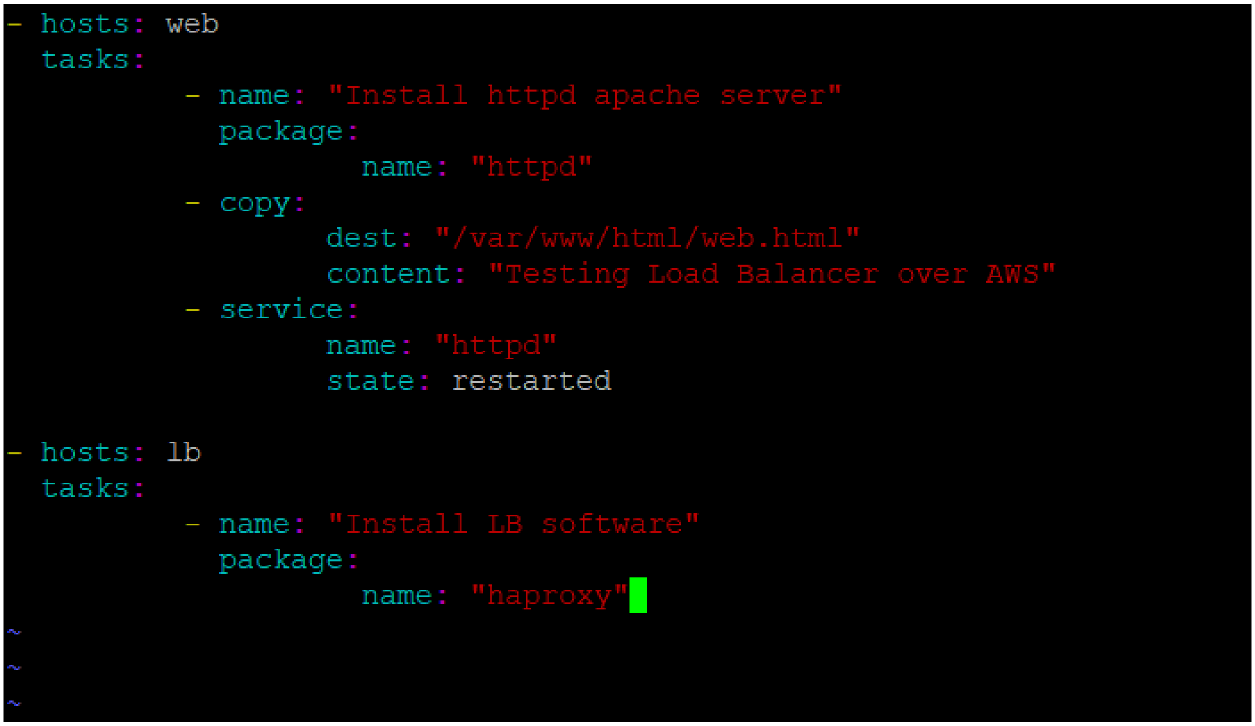
Do this step for both the target node. Now, check for the connectivity using **ansible all –m ping**



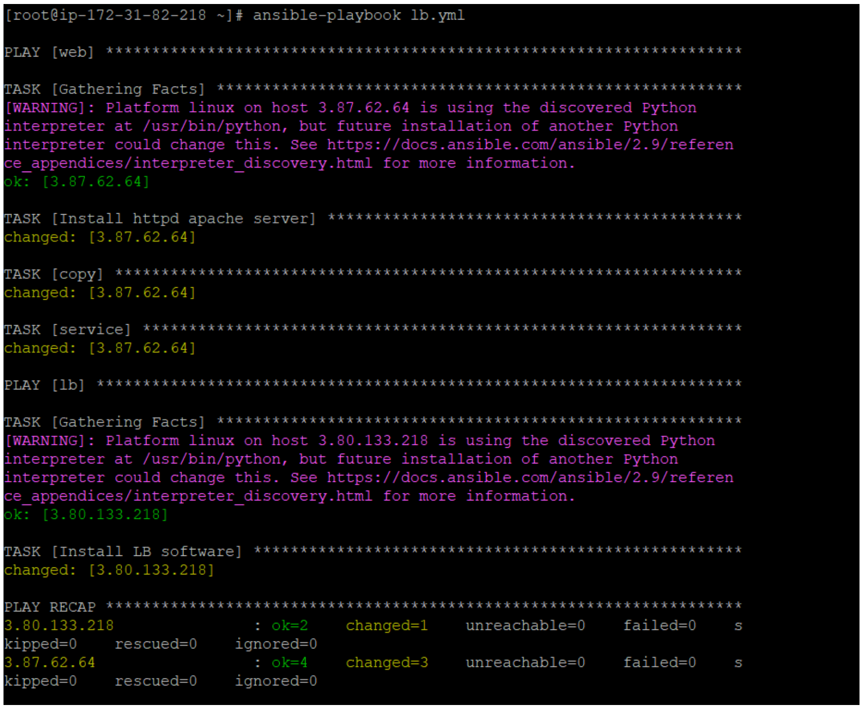
```
[root@ip-172-31-82-218 ~]# ansible all -m ping
The authenticity of host '3.80.133.218 (3.80.133.218)' can't be established.
ECDSA key fingerprint is SHA256:28VnpKnmLDWqUz3FJkvRMCxGBbdqGVTZzCZ6ImYG67w.
ECDSA key fingerprint is MD5:ba:7d:cd:3b:8a:40:07:c0:43:d9:7f:14:4f:ed:98:15.
Are you sure you want to continue connecting (yes/no)? [WARNING]: Platform linux
 on host 3.87.62.64 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/referen
ce_appendices/interpreter_discovery.html for more information.
3.87.62.64 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python"
    },
    "changed": false,
    "ping": "pong"
}
yes
[WARNING]: Platform linux on host 3.80.133.218 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/referen
ce_appendices/interpreter_discovery.html for more information.
3.80.133.218 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python"
    },
    "changed": false,
    "ping": "pong"
}
```

TASK:

Now, our inventory is good enough for further use. Our task is to set up a load balancer. For that, first we configure apache web-server on one target node and haproxy on the second.

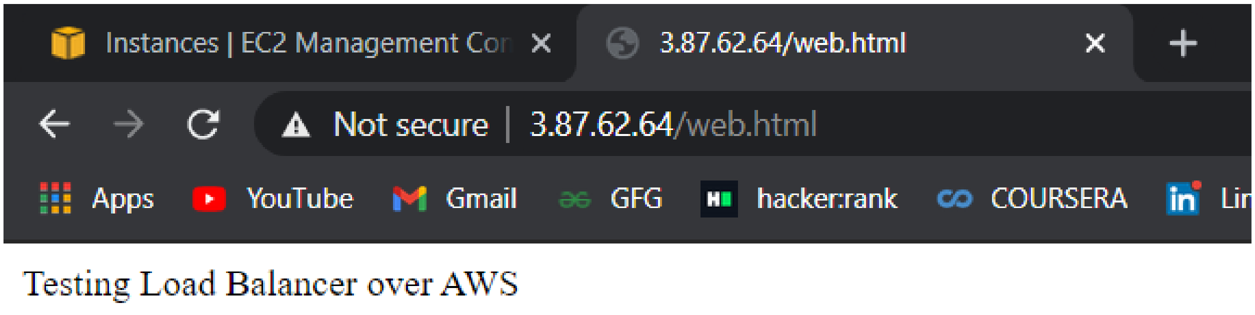


```
- hosts: web
  tasks:
          - name: "Install httpd apache server"
            package:
                    name: "httpd"
          - copy:
                  dest: "/var/www/html/web.html"
                  content: "Testing Load Balancer over AWS"
          - service:
                  name: "httpd"
                  state: restarted

- hosts: lb
  tasks:
          - name: "Install LB software"
            package:
                    name: "haproxy"
~
~
~
```



```
[root@ip-172-31-82-218 ~]# ansible-playbook lb.yml

PLAY [web] ********************************************************************

TASK [Gathering Facts] ********************************************************
[WARNING]: Platform linux on host 3.87.62.64 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/referen
ce_appendices/interpreter_discovery.html for more information.
ok: [3.87.62.64]

TASK [Install httpd apache server] ********************************************
changed: [3.87.62.64]

TASK [copy] *******************************************************************
changed: [3.87.62.64]

TASK [service] ****************************************************************
changed: [3.87.62.64]

PLAY [lb] *********************************************************************

TASK [Gathering Facts] ********************************************************
[WARNING]: Platform linux on host 3.80.133.218 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/referen
ce_appendices/interpreter_discovery.html for more information.
ok: [3.80.133.218]

TASK [Install LB software] ****************************************************
changed: [3.80.133.218]

PLAY RECAP ********************************************************************
3.80.133.218               : ok=2    changed=1    unreachable=0    failed=0    s
kipped=0    rescued=0    ignored=0
3.87.62.64                 : ok=4    changed=3    unreachable=0    failed=0    s
kipped=0    rescued=0    ignored=0
```

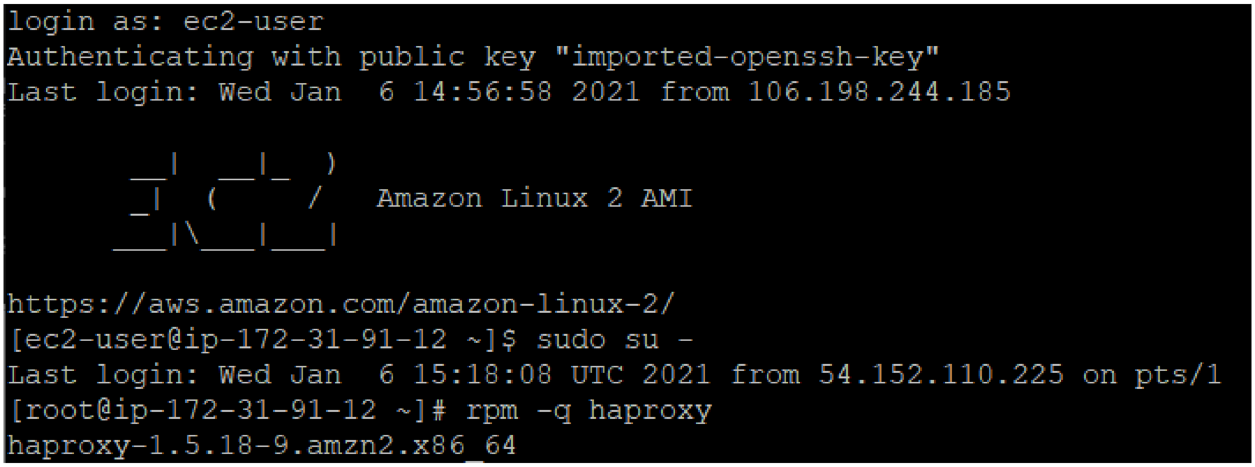
Go to the browser and give the ip of the webserver and name of the html file.



Testing Load Balancer over AWS

Now, our goal is to enable the service of haproxy on the other target node, so that we don’t need to reveal the IP of our webserver. We shall provide the IP of the loadbalancer only and users will reach to the page successfully.

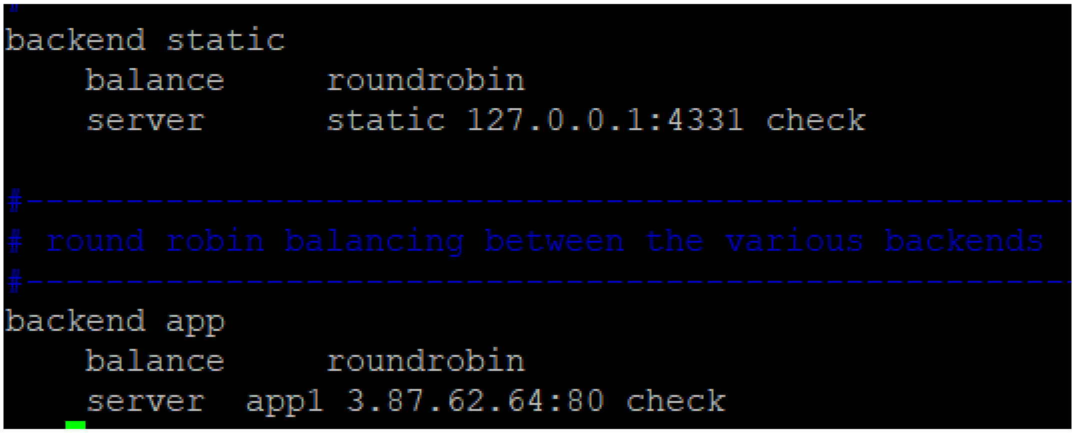
1. Check if haproxy is successfully installed on the target node using **rpm –q haproxy**



```
login as: ec2-user
Authenticating with public key "imported-openssh-key"
Last login: Wed Jan  6 14:56:58 2021 from 106.198.244.185

       __|  __|_  )
       _|  (     /   Amazon Linux 2 AMI
      ___|\___|___|

https://aws.amazon.com/amazon-linux-2/
[ec2-user@ip-172-31-91-12 ~]$ sudo su -
Last login: Wed Jan  6 15:18:08 UTC 2021 from 54.152.110.225 on pts/1
[root@ip-172-31-91-12 ~]# rpm -q haproxy
haproxy-1.5.18-9.amzn2.x86_64
```

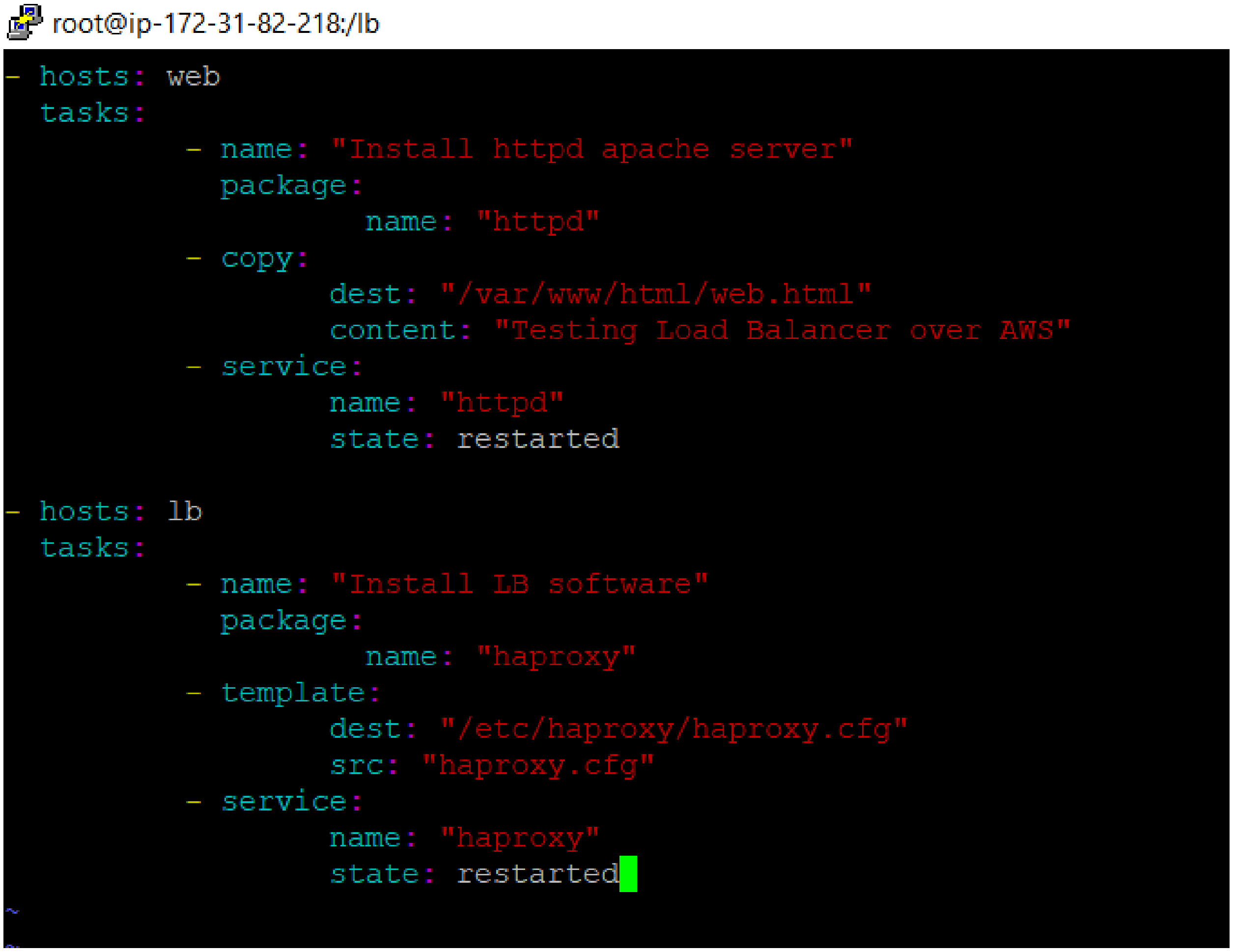
2. Copy config file of haproxy from target node to control node using **scp /etc/haproxy/haproxy.cfg ip_of_the_load_balancer:/name_of_the_directory**

3. Make the following changes in the haproxy.cfg file on the control node.



```
backend static
    balance     roundrobin
    server      static 127.0.0.1:4331 check

#-----------------------------------------------------
# round robin balancing between the various backends
#-----------------------------------------------------
backend app
    balance     roundrobin
    server   app1 3.87.62.64:80 check
```

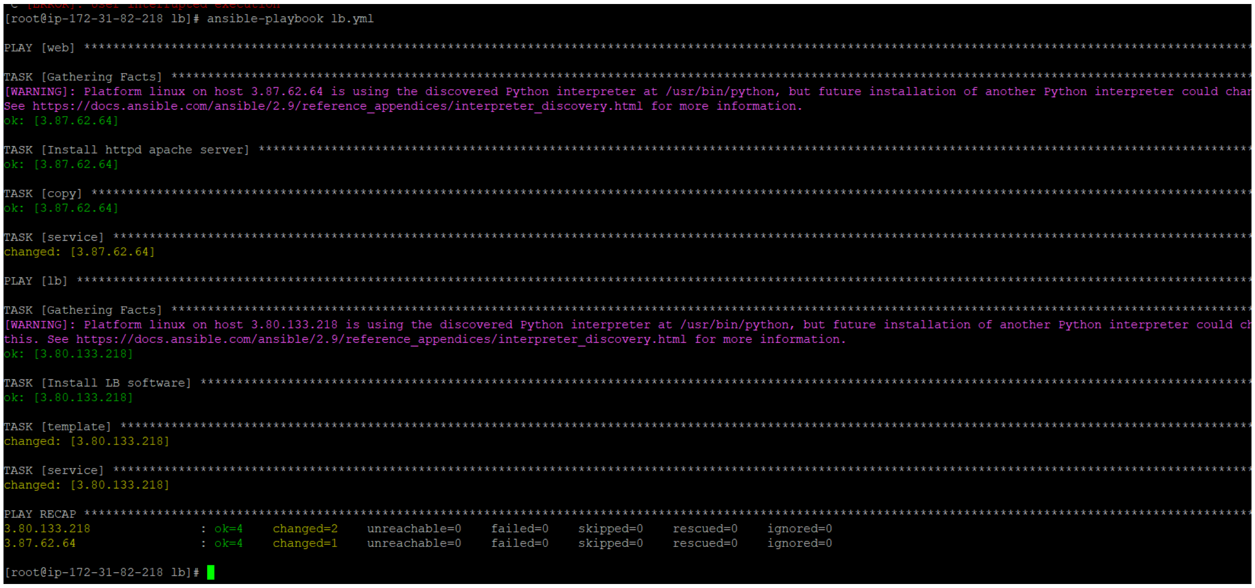
4. Write the following code in the yml file:



root@ip-172-31-82-218:/lb

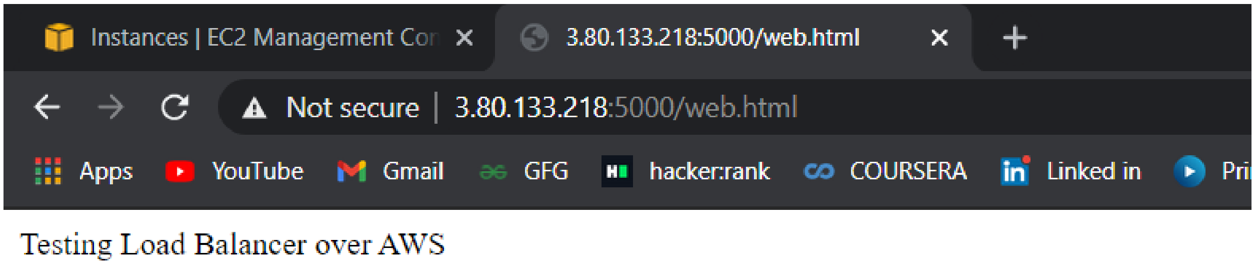
```
- hosts: web
  tasks:
          - name: "Install httpd apache server"
            package:
                    name: "httpd"
          - copy:
                  dest: "/var/www/html/web.html"
                  content: "Testing Load Balancer over AWS"
          - service:
                  name: "httpd"
                  state: restarted

- hosts: lb
  tasks:
          - name: "Install LB software"
            package:
                    name: "haproxy"
          - template:
                  dest: "/etc/haproxy/haproxy.cfg"
                  src: "haproxy.cfg"
          - service:
                  name: "haproxy"
                  state: restarted
~
```

5. Run the playbook



```
^C [ERROR]: User interrupted execution
[root@ip-172-31-82-218 lb]# ansible-playbook lb.yml

PLAY [web] ************************************************************************************************************************************

TASK [Gathering Facts] ************************************************************************************************************************
[WARNING]: Platform linux on host 3.87.62.64 is using the discovered Python interpreter at /usr/bin/python, but future installation of another Python interpreter could char
See https://docs.ansible.com/ansible/2.9/reference_appendices/interpreter_discovery.html for more information.
ok: [3.87.62.64]

TASK [Install httpd apache server] ************************************************************************************************************
ok: [3.87.62.64]

TASK [copy] ***********************************************************************************************************************************
ok: [3.87.62.64]

TASK [service] ********************************************************************************************************************************
changed: [3.87.62.64]

PLAY [lb] *************************************************************************************************************************************

TASK [Gathering Facts] ************************************************************************************************************************
[WARNING]: Platform linux on host 3.80.133.218 is using the discovered Python interpreter at /usr/bin/python, but future installation of another Python interpreter could ch
this. See https://docs.ansible.com/ansible/2.9/reference_appendices/interpreter_discovery.html for more information.
ok: [3.80.133.218]

TASK [Install LB software] ********************************************************************************************************************
ok: [3.80.133.218]

TASK [template] *******************************************************************************************************************************
changed: [3.80.133.218]

TASK [service] ********************************************************************************************************************************
changed: [3.80.133.218]

PLAY RECAP ************************************************************************************************************************************
3.80.133.218               : ok=4    changed=2    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
3.87.62.64                 : ok=4    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

[root@ip-172-31-82-218 lb]#
```

6. Go to the browser, type the ip of the load balancer, port number of haproxy, and name of the file.



3.80.133.218:5000/web.html

Testing Load Balancer over AWS

The task is successfully completed!