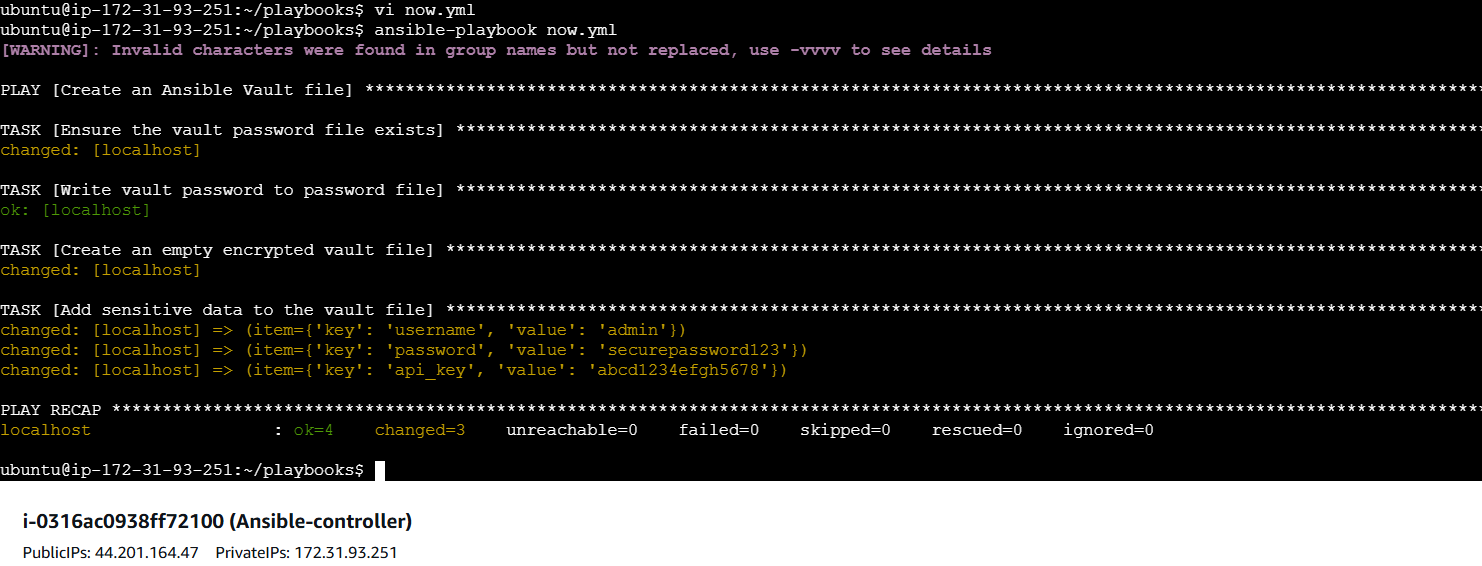
**Ansible Tasks-04:**

**1) Create ansible playbook to create ansible vault.**

|  |
| --- |
| ---  - name: Create an Ansible Vault file  hosts: localhost  gather\_facts: no  vars:  vault\_file\_path: "/home/ubuntu/playbooks/vault.yml"  vault\_password\_file: "/home/ubuntu/playbooks/vault\_password\_file"  sensitive\_data:  username: "admin"  password: "securepassword123"  api\_key: "abcd1234efgh5678"  tasks:  - name: Ensure the vault password file exists  ansible.builtin.file:  path: "{{ vault\_password\_file }}"  state: touch  mode: '0600'  - name: Write vault password to password file  ansible.builtin.copy:  dest: "{{ vault\_password\_file }}"  content: "my-vault-password\n"  owner: ubuntu  group: ubuntu  mode: '0600'  - name: Create an empty encrypted vault file  ansible.builtin.shell: >  ansible-vault encrypt --vault-password-file {{ vault\_password\_file }} --output {{ vault\_file\_path }} /dev/null  args:  executable: /bin/bash  - name: Add sensitive data to the vault file  ansible.builtin.shell: |  echo -e "{{ item.key }}: {{ item.value }}" | ansible-vault encrypt\_string --vault-password-file {{ vault\_password\_file }} >> {{ vault\_file\_path }}  loop: "{{ sensitive\_data | dict2items }}" |



**>>>Trying to cat the encrypted file:**



**2) Write a ansible playbook to install apache in linux and ubuntu machine by using when condition.**

**>>>The first script for the playbook which was throwing error:**

|  |
| --- |
| ---  - name: Install Apache on Linux and Ubuntu machines  hosts: all  become: yes  tasks:  - name: Install Apache on Ubuntu  apt: name=apache2 state=latest  when: ansible\_facts['distribution'] == "Ubuntu"  - name: Install Apache on Linux (other than Ubuntu)  yum: name=httpd state=installed  when: ansible\_facts['distribution'] == "Amazon"  - name: Start Apache service on Ubuntu  service:  name: apache2  state: started  enabled: yes  when: ansible\_facts['distribution'] == "Ubuntu"  - name: Start Apache service on Amazon Linux  service:  name: httpd  state: started  enabled: yes  when: ansible\_facts['distribution'] == "Amazon" |

**The error msg it was throwing:**



**To trouble shoot it:**

In the linux-server check for os-release

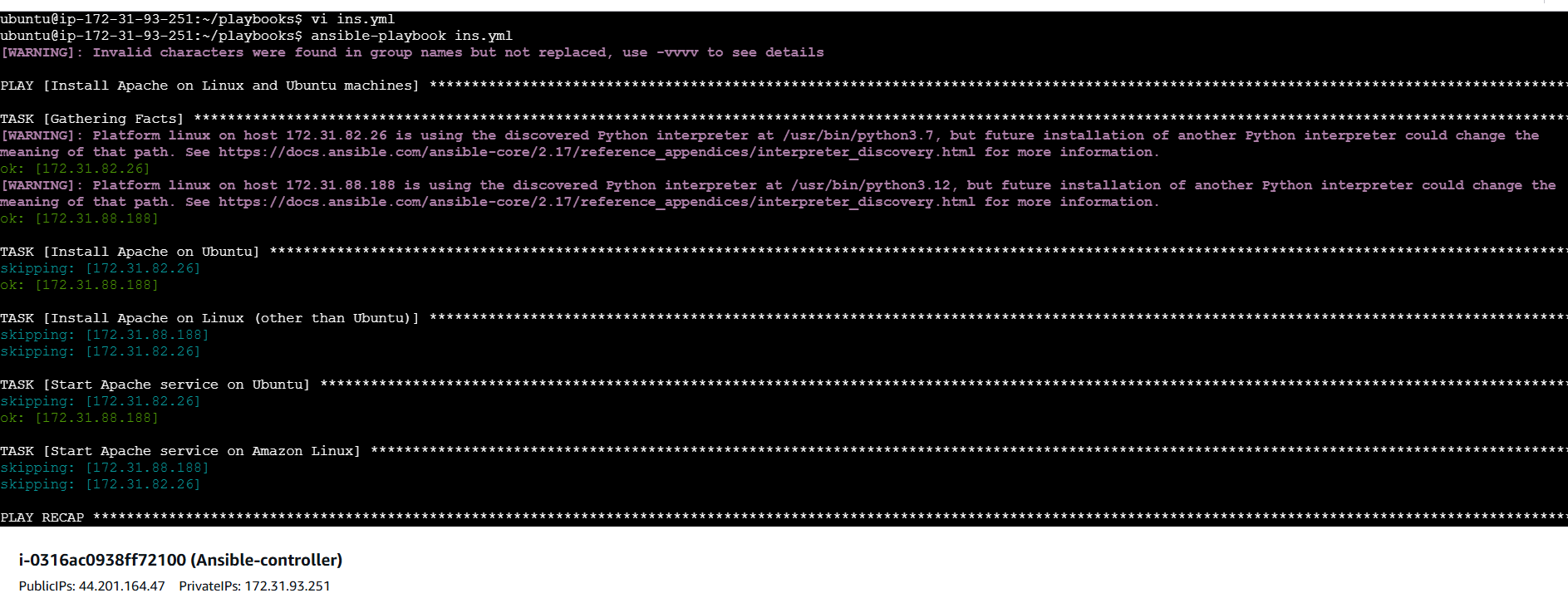
**cat /etc/os-release**

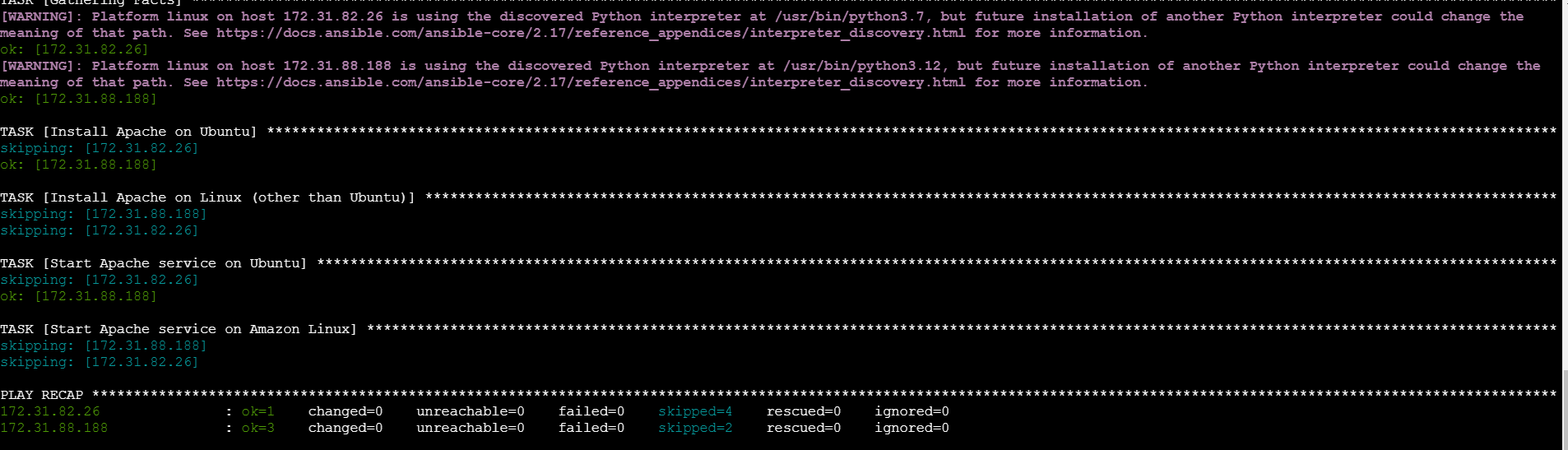
It turned out to be **NAME="Amazon Linux”**

All I had to do was to change the ansible facts [‘distributio’]==”Amazon” to ansible facts [‘distributio’]==”Amazon Linux”

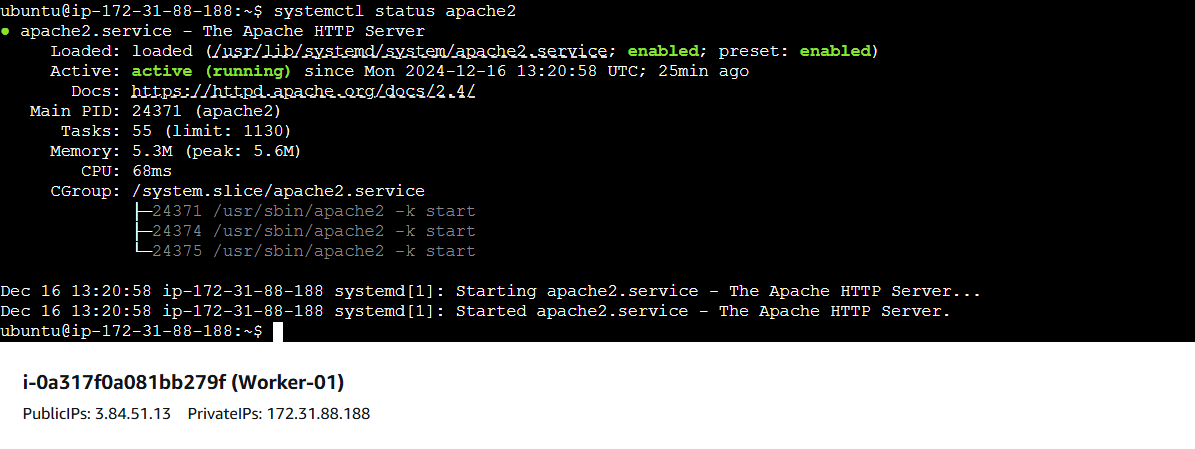
|  |
| --- |
| ---  - name: Install Apache on Linux and Ubuntu machines  hosts: all  become: yes  tasks:  - name: Install Apache on Ubuntu  apt: name=apache2 state=latest  when: ansible\_facts['distribution'] == "Ubuntu"  - name: Install Apache on Linux (other than Ubuntu)  yum: name=httpd state=installed  when: ansible\_facts['distribution'] == "Amazon Linux"  - name: Start Apache service on Ubuntu  service:  name: apache2  state: started  enabled: yes  when: ansible\_facts['distribution'] == "Ubuntu"  - name: Start Apache service on Amazon Linux  service:  name: httpd  state: started  enabled: yes  when: ansible\_facts['distribution'] == "Amazon Linux" |

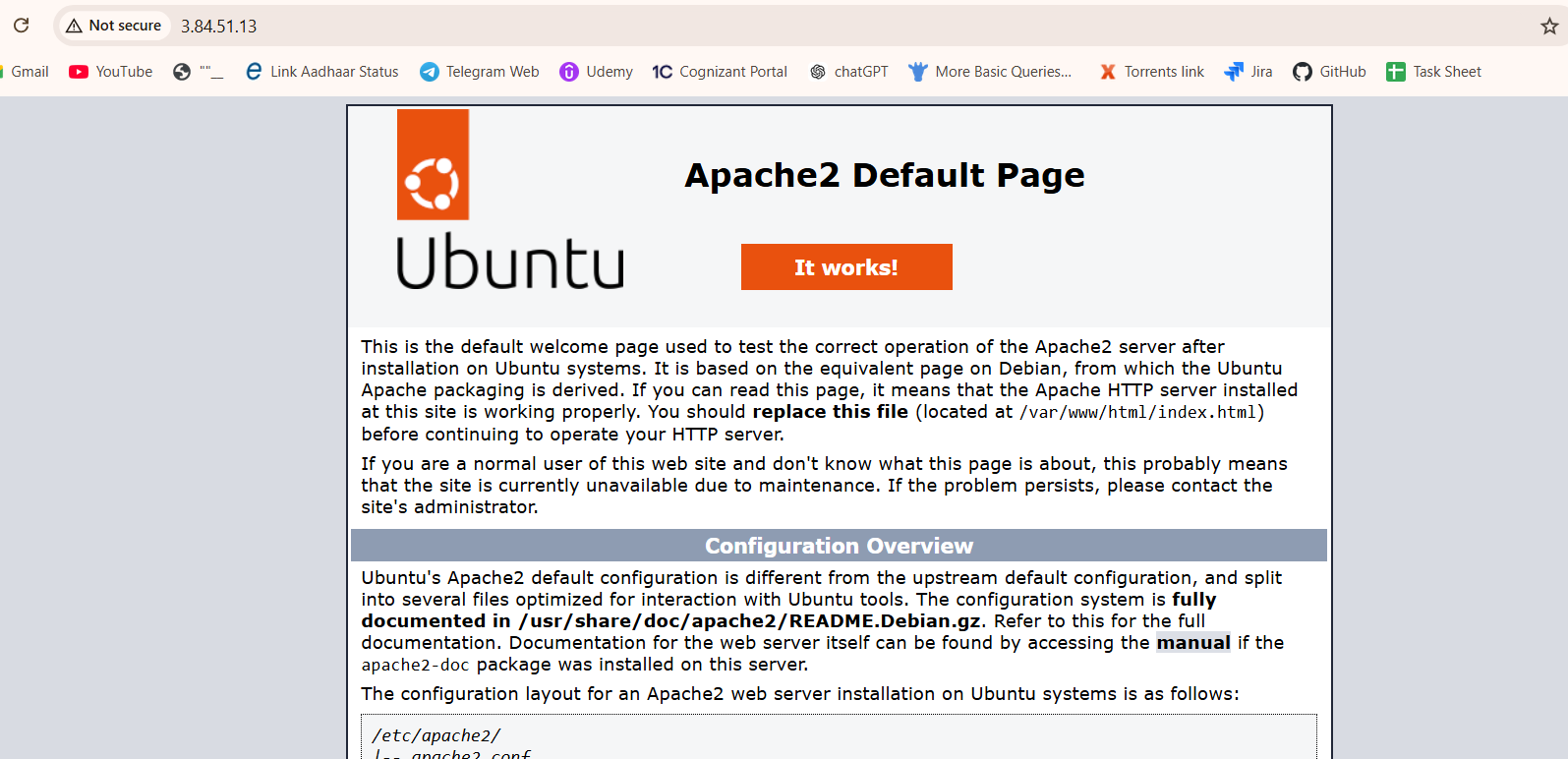
**The Updated Playbook:**



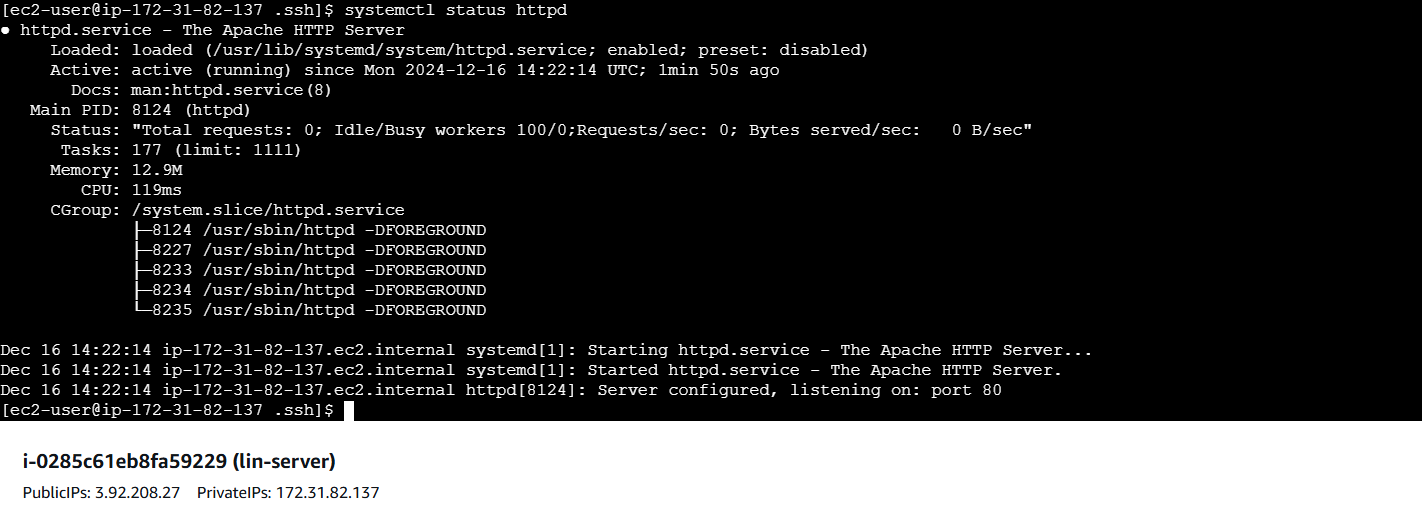


**Apache2 up and running on Ubuntu server:**





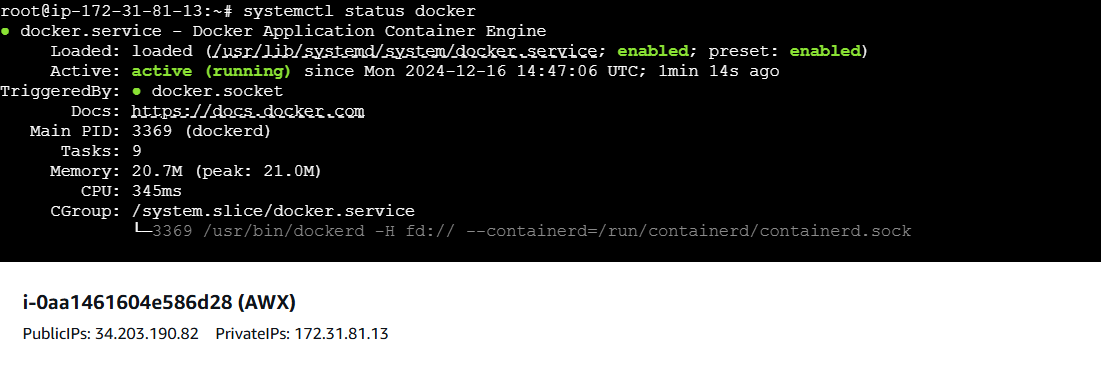
**Httpd up and running on Linux server:**



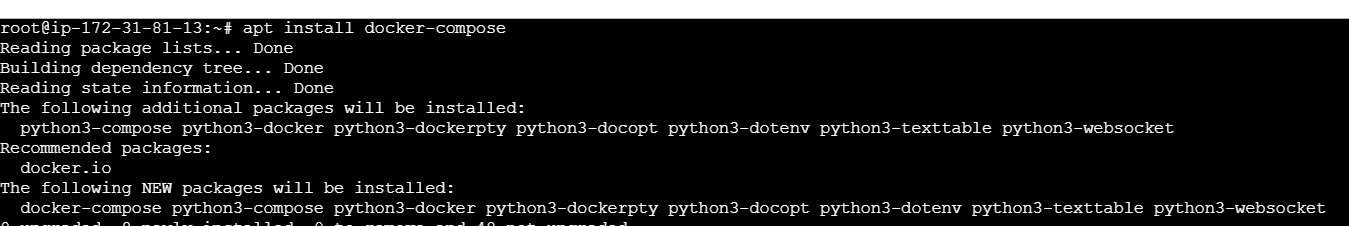


**3) Create ansible playbook using roles to configure LAMP stack**

**4) Setup ansible AWX and explore the options.**

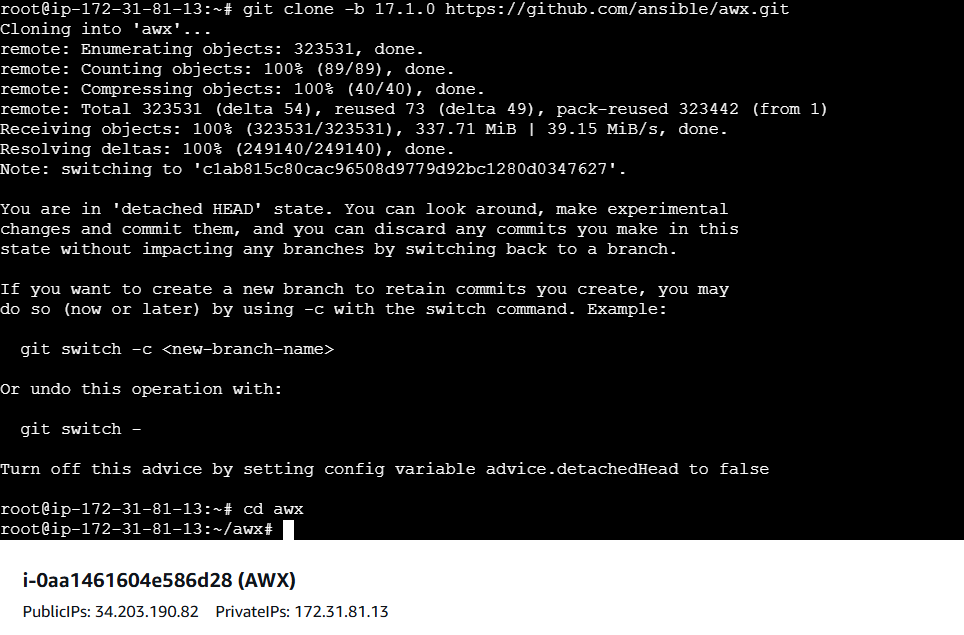
****

**Install docker-compose:**

****

**>>>Clone the repository:**

**git clone -b 17.1.0 https://github.com/ansible/awx.git**

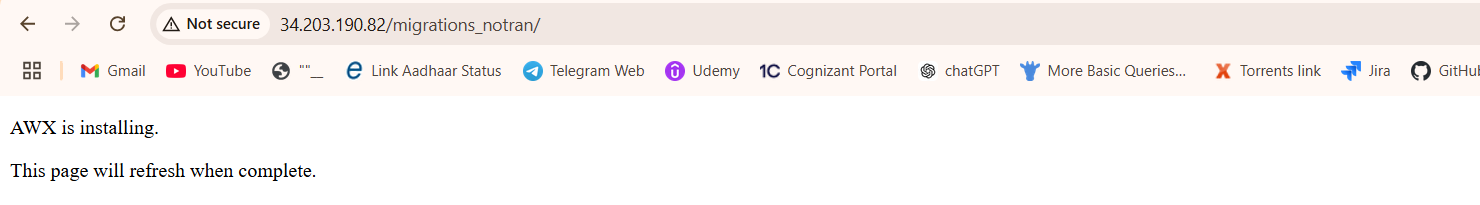
****

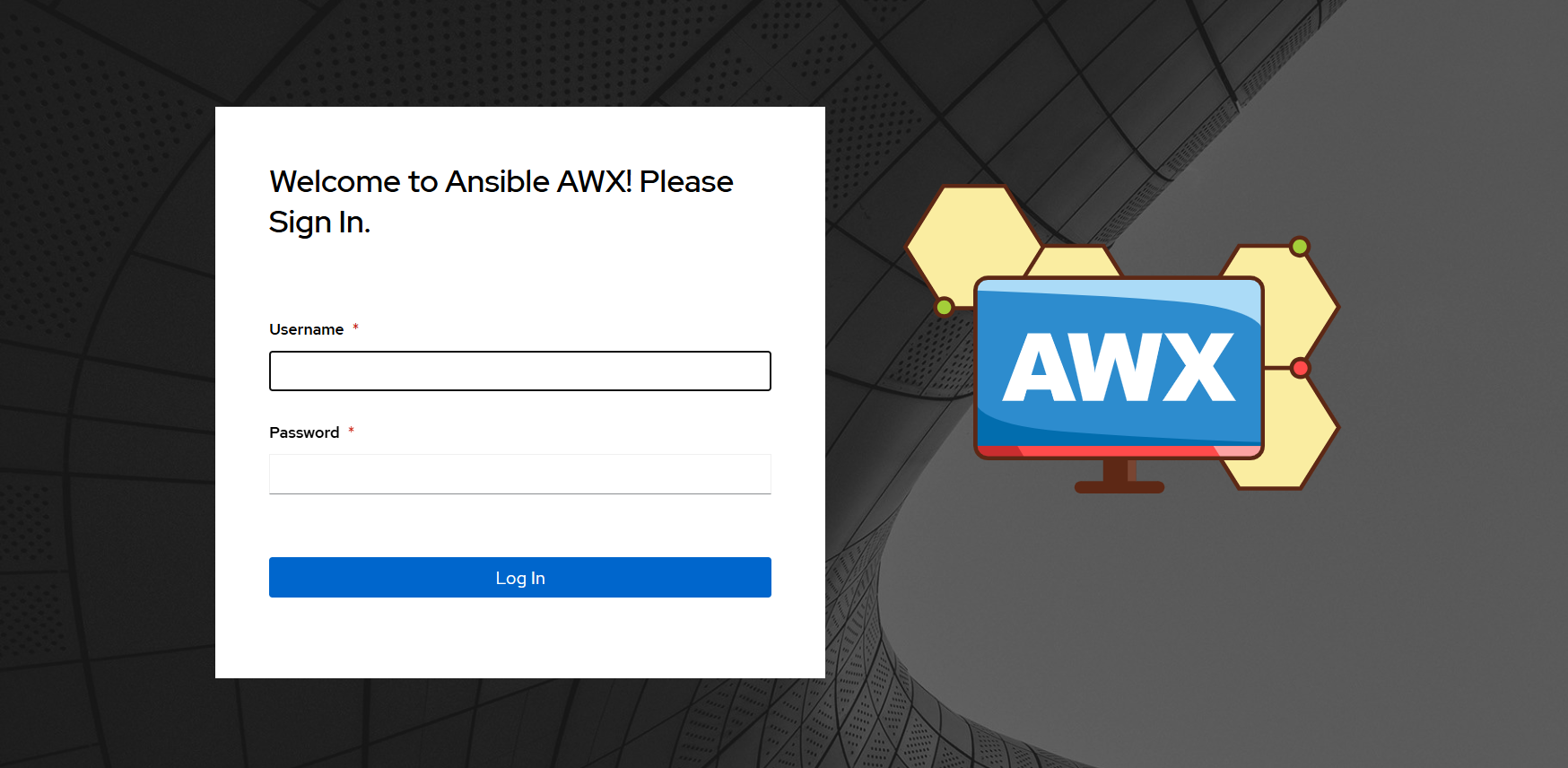
****

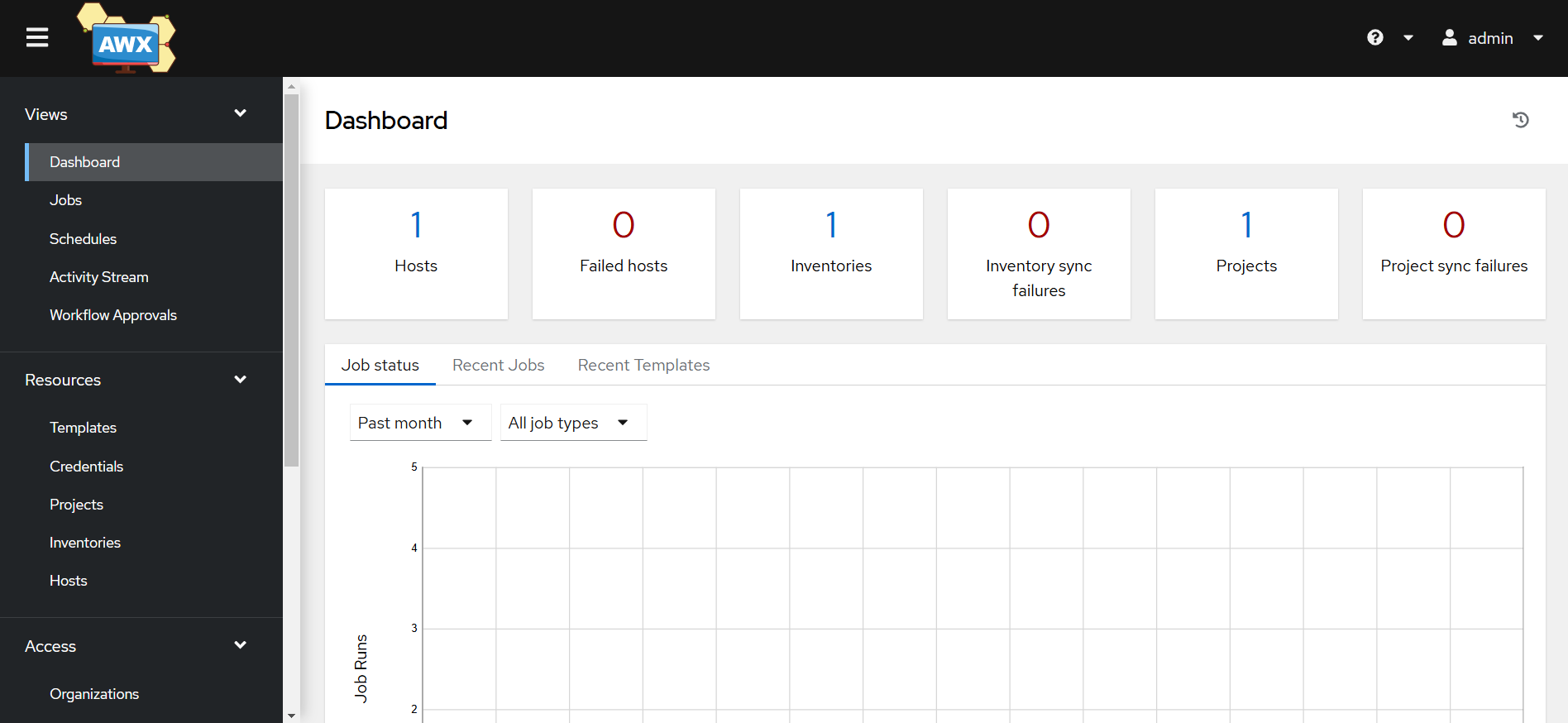
**Edit Inventory:**

admin\_password=Admin@123  
secret\_key=Admin@123  
pg\_database=awx  
pg\_password=Admin@123  
awx\_alternate\_dns\_servers="8.8.8.8,8.8.4.4"  
postgres\_data\_dir="/var/lib/awx/pgdocker"  
docker\_compose\_dir="/var/lib/awx/awxcompose"  
project\_data\_dir="/var/lib/awx/projects**"**

**Copy paste the public ip of the server on the browser:**

****

****

****