📌 Automate Kubernetes Cluster Using Ansible

⚙ Launch ec2-instances on AWS Cloud eg. for master and slave.

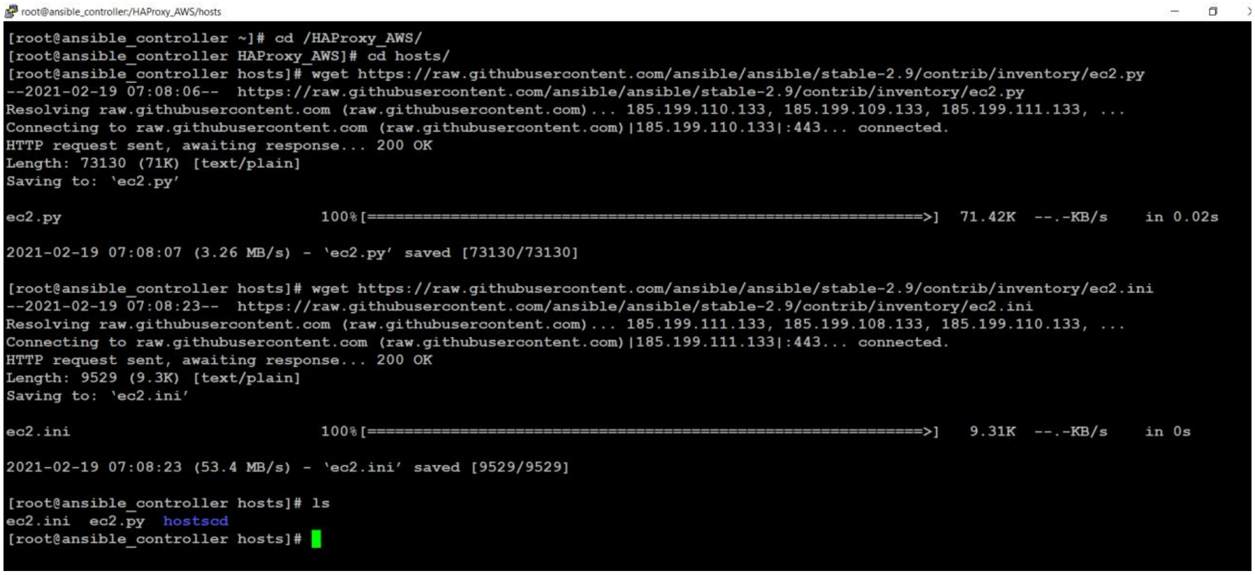
⚙ Create roles that will configure master node and slave node seperately.

⚙ Launch a wordpress and mysql database connected to it in the respectine slaves.

⚙ Expose the wordpress pod and client able hit the wordpress ip with its respective port.

Step1: Download ec2.py and ec2.ini using:

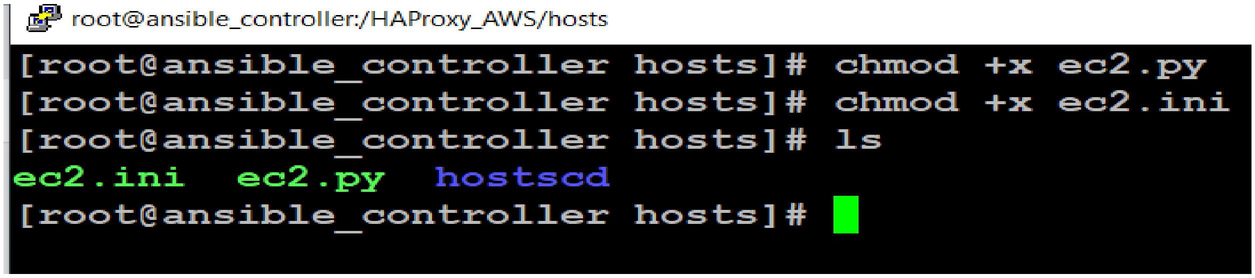
- wget https://raw.githubusercontent.com/ansible/ansible/stable-2.9/contrib/inventory/ec2.py
- wget https://raw.githubusercontent.com/ansible/ansible/stable-2.9/contrib/inventory/ec2.ini



```
[root@ansible_controller ~]# cd /HAProxy_AWS/
[root@ansible_controller HAProxy_AWS]# cd hosts/
[root@ansible_controller hosts]# wget https://raw.githubusercontent.com/ansible/ansible/stable-2.9/contrib/inventory/ec2.py
--2021-02-19 07:08:06--  https://raw.githubusercontent.com/ansible/ansible/stable-2.9/contrib/inventory/ec2.py
Resolving raw.githubusercontent.com (raw.githubusercontent.com)... 185.199.110.133, 185.199.109.133, 185.199.111.133, ...
Connecting to raw.githubusercontent.com (raw.githubusercontent.com)|185.199.110.133|: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 0.02s

2021-02-19 07:08:07 (3.26 MB/s) - 'ec2.py' saved [73130/73130]

[root@ansible_controller hosts]# wget https://raw.githubusercontent.com/ansible/ansible/stable-2.9/contrib/inventory/ec2.ini
--2021-02-19 07:08:23--  https://raw.githubusercontent.com/ansible/ansible/stable-2.9/contrib/inventory/ec2.ini
Resolving raw.githubusercontent.com (raw.githubusercontent.com)... 185.199.111.133, 185.199.108.133, 185.199.110.133, ...
Connecting to raw.githubusercontent.com (raw.githubusercontent.com)|185.199.111.133|:443... connected.
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

2021-02-19 07:08:23 (53.4 MB/s) - 'ec2.ini' saved [9529/9529]

[root@ansible_controller hosts]# ls
ec2.ini  ec2.py  hostscd
[root@ansible_controller hosts]# 
```

➔Make ec2.py and ec2.ini executable using:

- chmod +x ec2.py
- chmod +x ec2.ini

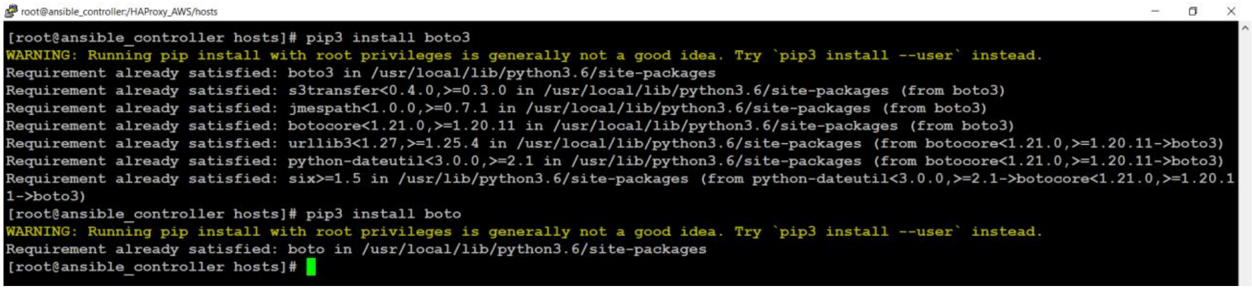


root@ansible_controller:/HAProxy_AWS/hosts

```
[root@ansible_controller hosts]# chmod +x ec2.py
[root@ansible_controller hosts]# chmod +x ec2.ini
[root@ansible_controller hosts]# ls
ec2.ini   ec2.py   hostscd
[root@ansible_controller hosts]# 
```

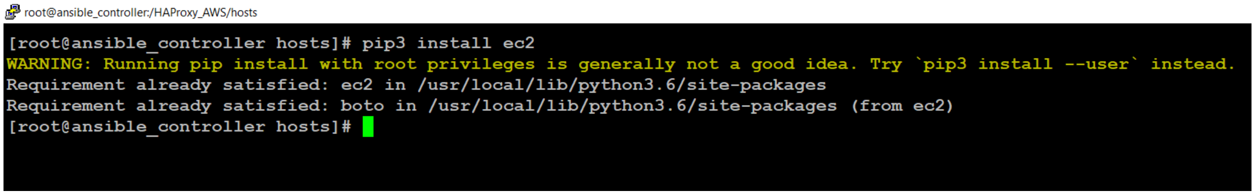
Step2: Download Boto3, Boto, ec2 packages using:

- pip3 install boto3
- pip3 install boto
- pip3 install ec2



root@ansible_controller:/HAProxy_AWS/hosts

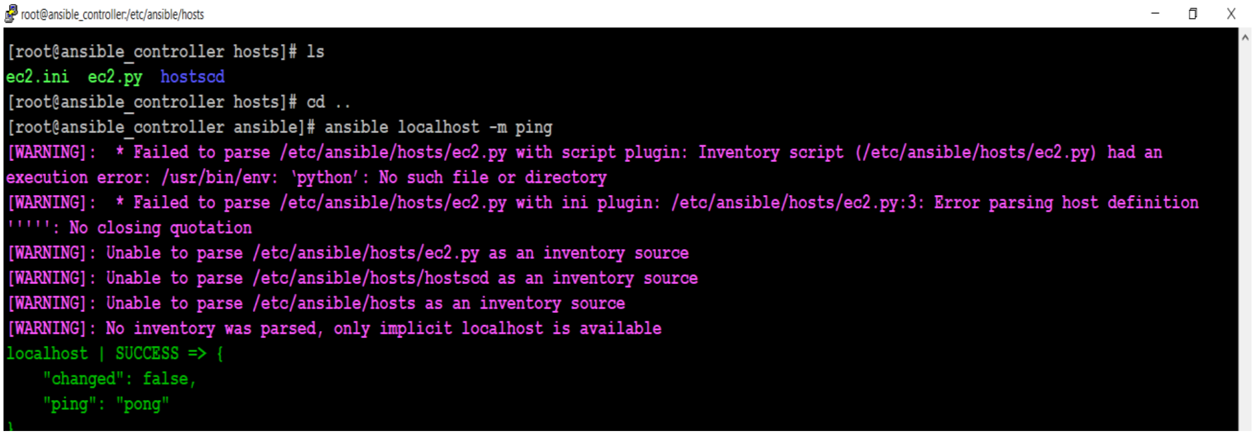
```
[root@ansible_controller hosts]# pip3 install boto3
WARNING: Running pip install with root privileges is generally not a good idea. Try `pip3 install --user` instead.
Requirement already satisfied: boto3 in /usr/local/lib/python3.6/site-packages
Requirement already satisfied: s3transfer<0.4.0,>=0.3.0 in /usr/local/lib/python3.6/site-packages (from boto3)
Requirement already satisfied: jmespath<1.0.0,>=0.7.1 in /usr/local/lib/python3.6/site-packages (from boto3)
Requirement already satisfied: botocore<1.21.0,>=1.20.11 in /usr/local/lib/python3.6/site-packages (from boto3)
Requirement already satisfied: urllib3<1.27,>=1.25.4 in /usr/local/lib/python3.6/site-packages (from botocore<1.21.0,>=1.20.11->boto3)
Requirement already satisfied: python-dateutil<3.0.0,>=2.1 in /usr/lib/python3.6/site-packages (from botocore<1.21.0,>=1.20.11->boto3)
Requirement already satisfied: six>=1.5 in /usr/lib/python3.6/site-packages (from python-dateutil<3.0.0,>=2.1->botocore<1.21.0,>=1.20.1
1->boto3)
[root@ansible_controller hosts]# pip3 install boto
WARNING: Running pip install with root privileges is generally not a good idea. Try `pip3 install --user` instead.
Requirement already satisfied: boto in /usr/local/lib/python3.6/site-packages
[root@ansible_controller hosts]# 
```



root@ansible_controller:/HAProxy_AWS/hosts

```
[root@ansible_controller hosts]# pip3 install ec2
WARNING: Running pip install with root privileges is generally not a good idea. Try `pip3 install --user` instead.
Requirement already satisfied: ec2 in /usr/local/lib/python3.6/site-packages
Requirement already satisfied: boto in /usr/local/lib/python3.6/site-packages (from ec2)
[root@ansible_controller hosts]# 
```

➔Check the dynamic inventory is working fine or not.
➔You will get these kinds of warnings.

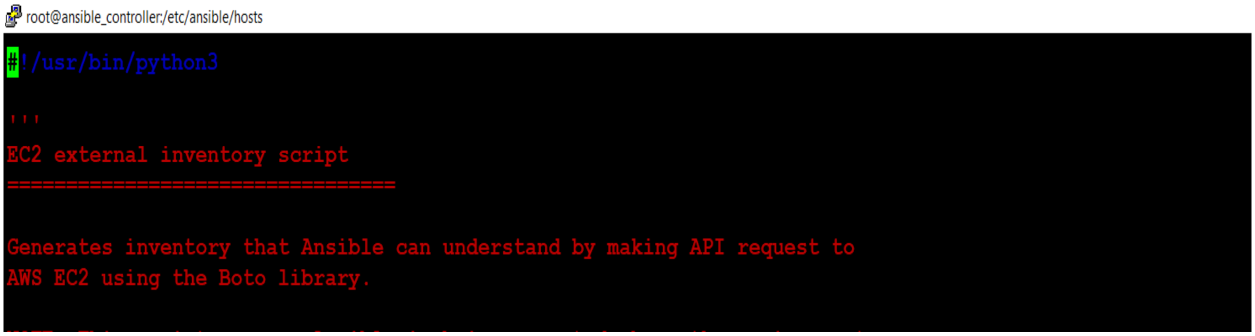


root@ansible_controller:/etc/ansible/hosts

```
[root@ansible_controller hosts]# ls
ec2.ini  ec2.py  hostscd
[root@ansible_controller hosts]# cd ..
[root@ansible_controller ansible]# ansible localhost -m ping
[WARNING]:  * Failed to parse /etc/ansible/hosts/ec2.py with script plugin: Inventory script (/etc/ansible/hosts/ec2.py) had an
execution error: /usr/bin/env: 'python': No such file or directory
[WARNING]:  * Failed to parse /etc/ansible/hosts/ec2.py with ini plugin: /etc/ansible/hosts/ec2.py:3: Error parsing host definition
''''': No closing quotation
[WARNING]: Unable to parse /etc/ansible/hosts/ec2.py as an inventory source
[WARNING]: Unable to parse /etc/ansible/hosts/hostscd as an inventory source
[WARNING]: Unable to parse /etc/ansible/hosts as an inventory source
[WARNING]: No inventory was parsed, only implicit localhost is available
localhost | SUCCESS => {
    "changed": false,
    "ping": "pong"
}
```

➔ To solve this Warnings

- Open ec2.py, In line1 Change
  #!/usr/bin/env python → #!/usr/bin/python3

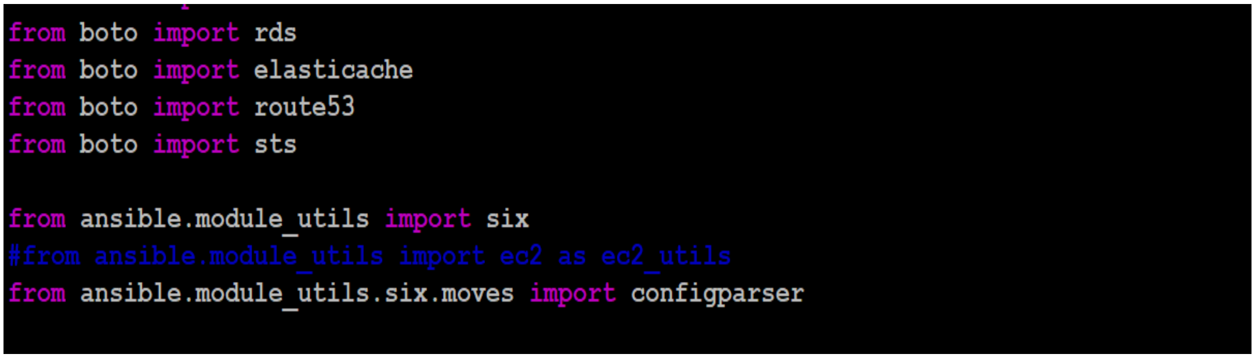


root@ansible_controller:/etc/ansible/hosts

```
#!/usr/bin/python3

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

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

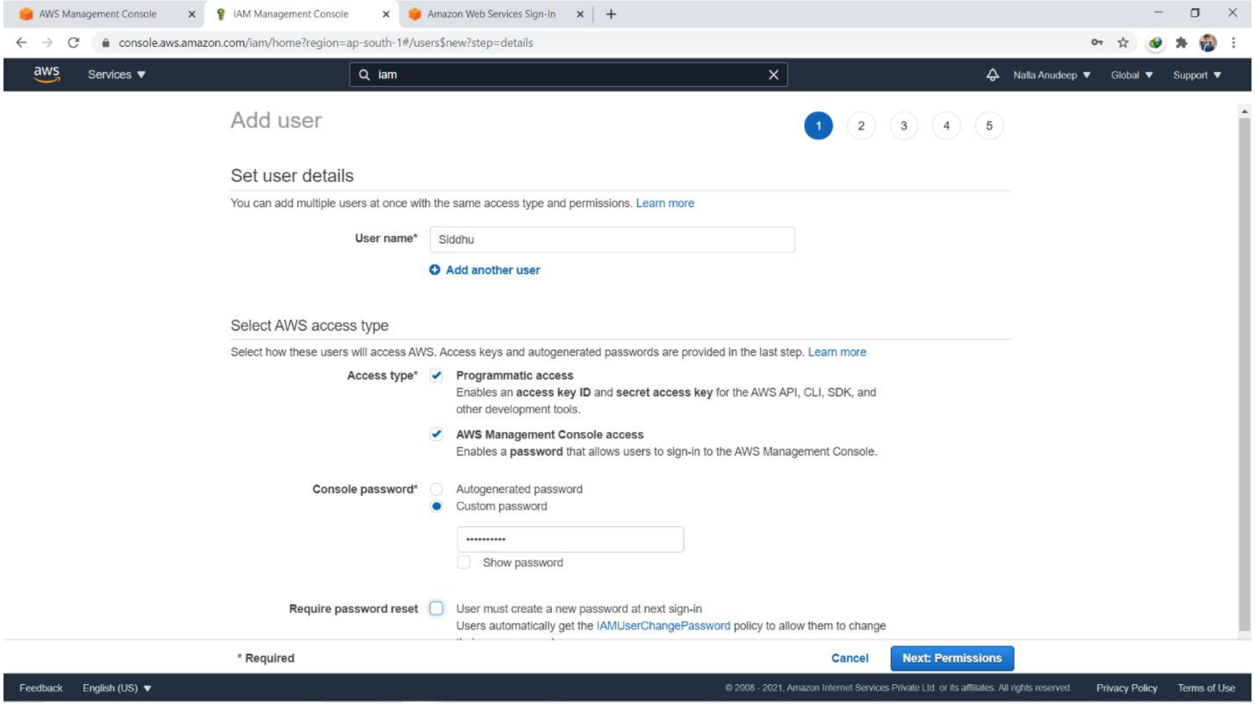
➔Comment the Line 172 in ec2.py i.e.,

- from ansible.module_utils import ec2 as ec2_utils



```
from boto import rds
from boto import elasticache
from boto import route53
from boto import sts

from ansible.module_utils import six
#from ansible.module_utils import ec2 as ec2_utils
from ansible.module_utils.six.moves import configparser
```

Step3: Create the IAM user and export Access key and Secret key of user.



AWS Management Console
IAM Management Console
Amazon Web Services Sign-In
console.aws.amazon.com/iam/home?region=ap-south-1#/users$new?step=details
aws
Services ▼
iam
Nalla Anudeep ▼
Global ▼
Support ▼

Add user

1 2 3 4 5

Set user details

You can add multiple users at once with the same access type and permissions. Learn more

User name* Siddhu

Add another user

Select AWS access type

Select how these users will access AWS. Access keys and autogenerated passwords are provided in the last step. Learn more

Access type* ☑ **Programmatic access**
Enables an **access key ID** and **secret access key** for the AWS API, CLI, SDK, and other development tools.

☑ **AWS Management Console access**
Enables a **password** that allows users to sign-in to the AWS Management Console.

Console password* ○ Autogenerated password
● Custom password

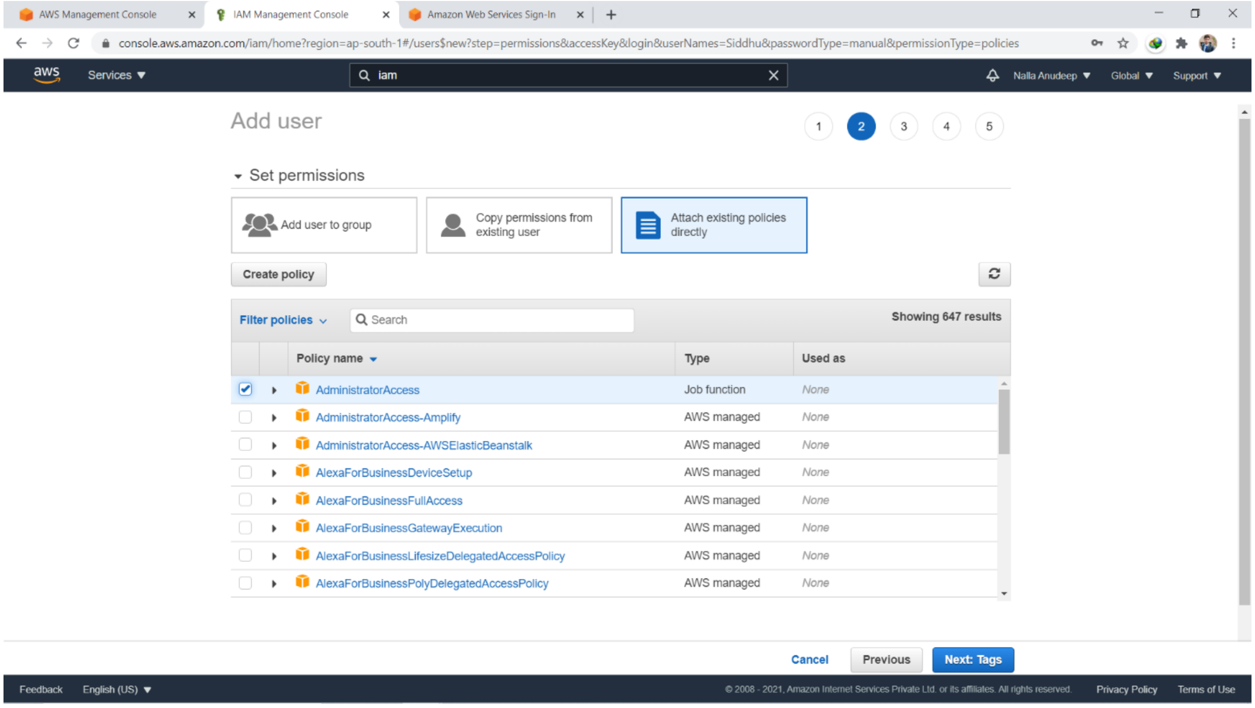
••••••••••

□ Show password

Require password reset □ User must create a new password at next sign-in
Users automatically get the IAMUserChangePassword policy to allow them to change

* Required

Cancel
Next: Permissions
Feedback
English (US) ▼
© 2008 - 2021, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved.
Privacy Policy
Terms of Use



AWS Management Console
IAM Management Console
Amazon Web Services Sign-In
console.aws.amazon.com/iam/home?region=ap-south-1#/users$new?step=permissions&accessKey&login&userNames=Siddhu&passwordType=manual&permissionType=policies
aws
Services ▼
iam
Nalla Anudeep ▼
Global ▼
Support ▼

Add user

1 2 3 4 5

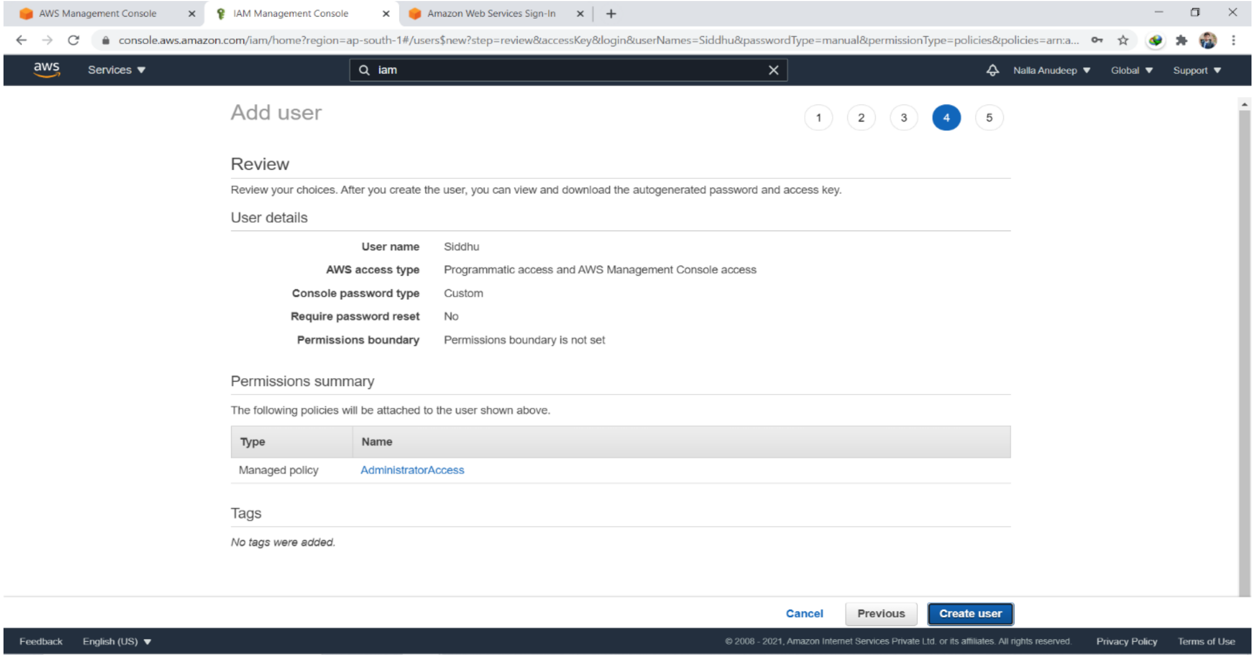
Set permissions

Add user to group
Copy permissions from existing user
Attach existing policies directly
Create policy
Filter policies
Search

Showing 647 results

		Policy name	Type	Used as
☑	▸	AdministratorAccess	Job function	None
□	▸	AdministratorAccess-Amplify	AWS managed	None
□	▸	AdministratorAccess-AWSElasticBeanstalk	AWS managed	None
□	▸	AlexaForBusinessDeviceSetup	AWS managed	None
□	▸	AlexaForBusinessFullAccess	AWS managed	None
□	▸	AlexaForBusinessGatewayExecution	AWS managed	None
□	▸	AlexaForBusinessLifesizeDelegatedAccessPolicy	AWS managed	None
□	▸	AlexaForBusinessPolyDelegatedAccessPolicy	AWS managed	None

Cancel
Previous
Next: Tags
Feedback
English (US) ▼
© 2008 - 2021, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved.
Privacy Policy
Terms of Use


AWS Management Console
IAM Management Console
Amazon Web Services Sign-In
console.aws.amazon.com/iam/home?region=ap-south-1#/users$new?step=review&accessKey&login&userNames=Siddhu&passwordType=manual&permissionType=policies&policies=arn:a...
aws
Services ▼
iam
Nalla Anudeep ▼
Global ▼
Support ▼
1
2
3
4
5
Cancel
Previous
Create user
Feedback
English (US) ▼
© 2008 - 2021, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved.
Privacy Policy
Terms of Use

Add user

Review

Review your choices. After you create the user, you can view and download the autogenerated password and access key.

User details

User name Siddhu

AWS access type Programmatic access and AWS Management Console access

Console password type Custom

Require password reset No

Permissions boundary Permissions boundary is not set

Permissions summary

The following policies will be attached to the user shown above.

Type	Name
Managed policy	AdministratorAccess

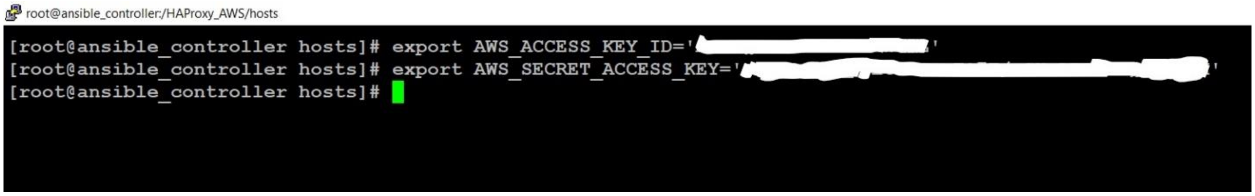
Tags

No tags were added.

The keyword used to set key is –

→ export AWS_ACCESS_KEY_ID='<Your_Access_key>'

→ export AWS_SECRET_ACCESS_KEY='<Your_Secret_Key>'


root@ansible_controller:/HAProxy_AWS/hosts

```
[root@ansible_controller hosts]# export AWS_ACCESS_KEY_ID='[redacted]'
[root@ansible_controller hosts]# export AWS_SECRET_ACCESS_KEY='[redacted]'
[root@ansible_controller hosts]#
```

Step4: Configure the ansible.cfg file:

[defaults]

inventory = /etc/ansible/hosts

```
host_key_checking = False
deprecation_warnings = False
remote_user = ec2-user
private_key_file = /etc/ansible/ansiblekey.pem
roles_path = /etc/ansible/task23/roles
command_warnings = False
ask_pass = False
interpreter_python = /usr/bin/python
[privilege_escalation]
become = true
become_method = sudo
become_user = root
become_ask_pass = false
```

Check The Code From GitHub

GitHub link: https://github.com/Anuddeeph/Configure_k8s_Using_Ansible_Separately.git

Implementation Video Drive Link:
https://drive.google.com/file/d/1ydL1GTLRHr1wJ28OIvd5UL3F0I_ybHLb/view?usp=sharing