**Step1:**

* Login to ur EC2 instance using .pem
* Change the user to root
* Create a new user lucky(for my case)…. Ex **useradd <user\_name>**
* Set the password for lucky user (lucky123) …. Ex **passwd <user\_name>**

**TYPE password: <your\_password>**

**Retype password: <Retype\_same>**

* New user has been created.

**Step2:**

* Check whether Password Authentication and permit root login enabled for **root** user?
* Go to **/etc/ssh directory**
* Open **sshd\_config** file.
* **Vi sshd\_config** and un comment the below items and save the file.

PasswordAuthentication



And Permit root login



* Do the same all the slave servers.
* Restart the serveces. (**systemctl restart sshd**)

**Step3:**

Give root privilege to the test user <lucky>.

Open file using command  then add the following lines

lucky ALL=(ALL) NOPASSWD: ALL

**Step4 : to enable password less communication**

* Generate a key of master node using below command.



* ssh-copy-id <private\_ip\_of\_ other\_machine>
* its gonna ask for the password for the first time, then afterwards its not going to ask any password
* do the same in slave nodes also.
* Also form slave nodes to slave nodes… so that we can able to connect from any node to any node.

**Step5 : Deployment of ansible**

* 2 versions of ansible are available:

1. Ansible tower (with GUI- enterprise edition)
2. Free versions of ansible

* Before deployment of ansible, make sure EPEL is deployed in all machine(master and slaves)
* EPEL is the extra package release for redhat. Not need to install EPEL for AMAZON-LINUX-AMI. As its already present in the repo list.
* Do **sudo yum repolist,** while trying the same if its unable to connect the CDS load balancer, then **reboot** ur EC2 instance.
* Cd etc/yum.repos.d then do a ls. The list of all the repos will be listed.
* As you can see no EPEL repo is found, so our first work will be installing EPEL repo.
* Type wget http://dl.fedoraproject.org/ pub/epel/7/x84\_64/e/epel-release-7-7.noarch.rpm
* If in any case wget is not found then we need to install wget first.



* Go to lucky home directory.
* Then type the following command

wget <https://dl.fedoraproject.org/pub/epel/7/x86_64/e/epel-release-7-9.noarch.rpm>

* Then execute the below command

sudo rpm –ivh epel-release-7-9.noarch.rpm

* Now the epel repo will get created.

cd /etc/yum.repos.d

Ls

* Similarly configure the same in each and every machine(master and slave)
* We need to update yum.(make sure you are in lucky user)

 (master and all the slaves)

* NEXT we are gonna install ansible using below command.

sudo yum install ansible –y

* If ansible is not getting installed go to cd /etc/yum.repos.d directory and edit the epel.repo file and make enable 1.
* We need to make sure ansible.cfg file is present under etc/ansible directory.
* If you want to enable login,go into ansible.cfg and uncomment the line



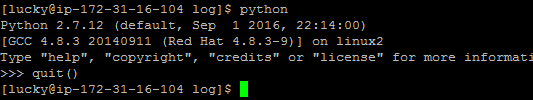
* Change the permisiion before making changes.
* Create a file called anisble.log() in the following directory (/var/log)and also give 777 permission.

**Step5 : Configuration of ansible in master node:**

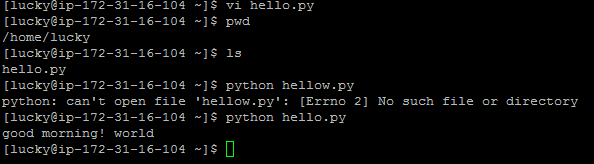
* Its better to a back up of of the ansible configuration file before making any changes init.
* Almost all the parameter of the cfg file can be over-ridden in the ansible playbook.
* Ansible will read the Ansible\_Config from the current woring directory, ansible.cfg from the home directory or /etc/ansible/ansible.cfg, which ever comes first.
* There a whole bunch of tags are there. Must to know the work of all the tags.

**Step6 : Check python installation**

* By default python gets installed with the ansible.
* Sudo yum installed|grep python.
* Check python. Just type python (enter)



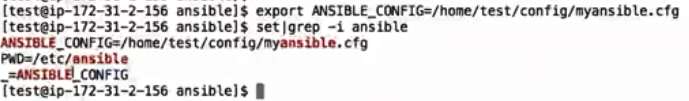
* Write a simple hellow world python program and test it.



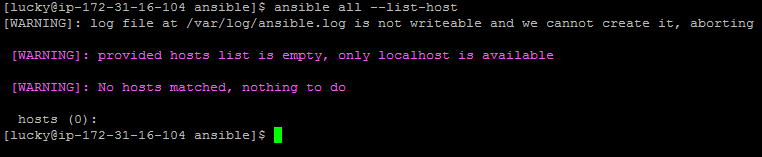
**Step 7: Overriding default ansible configuration**

* Generally ansible picks the ansible.cfg file from the **(/etc/ansible) .**
* To check present working directory . Use command **set|grep –i Ansible.**
* Its basically like environmental variable in windows system, if you don’t want ansible to pick the configuration file from the mention location, you can definitely change the same.

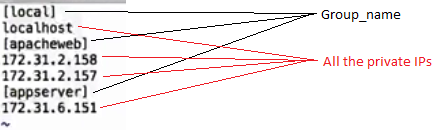
Ex:

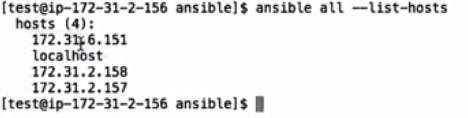


* Overriding ansible config,rolls path and host file (also take a back up of this file)
* It contains all the host details.
* Before making any changes we need to execute the following command to know what are the host files present.

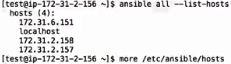
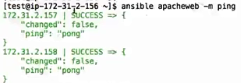
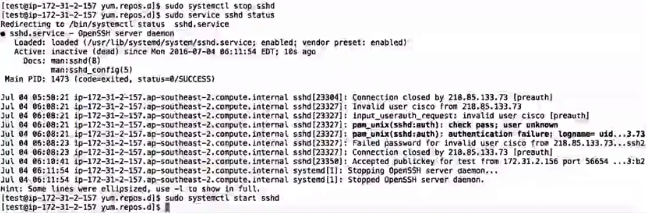
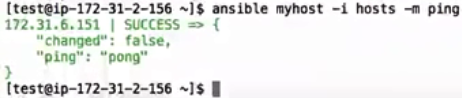


* We are going to add hosts in that file. Hosts are nothing,but the nodes.
* Make sure when you are working internally, the ip details you provide, should be a private IP.
* Do a **vi** to the file and delete all the pre defined details.
* The first node you add in the host file is your local ip.(i.e master ip) then the Node/child IP.



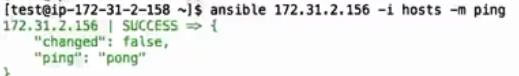


* **Q. How to override host file?**

1. **The original host file is present inside /etc/ansible/hosts directory. As per ansible.cfg,the above location is default location. In the home directory I am going to create a file called hosts and provided some details, Still ansible is gonna pick the original file.**
2. **Lets test the configuration from one node to all the other node.**
3. **Check whether you have all the hosts available or not** 
4. **And all the hosts listed in original host file** 
5. **Checking connectivity** 
6. **Stopping and checking the status and restarting**
7. **To override the default host file, we need to use below command.** 

**OR**

**Create a host file in home directory add any host ip, and to override use the below command**



* **Q. How to convert one node to a master node.**

1. **Login in that machine and edit the hosts file delete all the content and add all the other node details. Mark the login node local host Ex**



**Note: Ansible reads the hosts file from top to down approach.**

**How to replace a lost secret access key?**

Follow these simple steps:

**Step 1:** Create a new access key, which includes a new secret access key.

* To create a new secret access key for your root account, use the [security credentials page](https://console.aws.amazon.com/iam/home?#security_credential). Expand the **Access Keys** section, and then click **Create New Root Key**.
* To create a new secret access key for an IAM user, open the [IAM console](https://console.aws.amazon.com/iam/home?region=us-east-1#home). Click **Users** in the **Details** pane, click the appropriate IAM user, and then click **Create Access Key** on the **Security Credentials** tab.

**Note:** If you already have the maximum of two access keys—active or inactive—you must delete one first before proceeding. If you need more than two root access keys, [IAM users](http://docs.aws.amazon.com/IAM/latest/UserGuide/id_users.html) (each of whom can be assigned their own access keys) would probably better suit your requirements.

**Step 2:** Download the newly created credentials, when prompted to do so in the key creation wizard./OR you can copy both the keys,but better download

**Step 3:** Make your unused access keys inactive in case you need to roll back, and then delete them when you’re sure that they’re no longer needed (after you have confirmed that they are not in use by any of your applications). The [access key last used feature](https://blogs.aws.amazon.com/security/post/Tx1GZCHQC7LR3UT/New-in-IAM-Quickly-Identify-When-an-Access-Key-Was-Last-Used) can help you validate if keys are still in use.

**Note:** Access keys in an “Inactive” state still count toward your maximum of two access keys at any given time.-