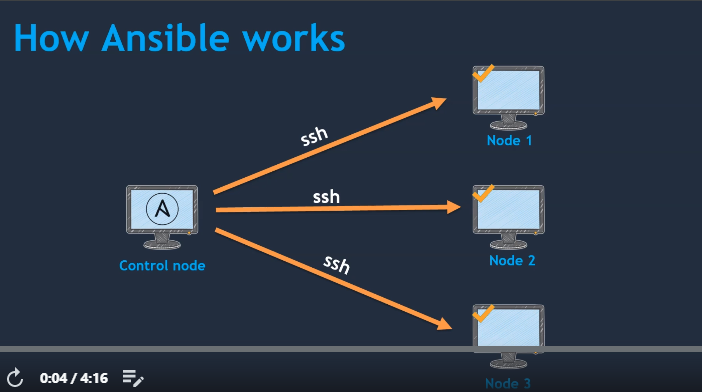
**Configuration Management**: Configuration management is a process for maintaining computer systems, servers, and software in a desired, consistent state. It’s a way to make sure that a system performs as it’s expected to as changes are made over time.

Ansible is an IT automation tool. It can configure systems, deploy software, and orchestrate more advanced IT tasks such as continuous deployments or zero downtime rolling updates.

Ansible is an agentless automation tool that you install on a control node. From the control node, Ansible manages machines and other devices remotely (by default, over the SSH protocol).



In Control node we will install ansible so wherever we install anisble is called as control nodes and as it is agentless automation tool we no need to install anisble on other nodes.

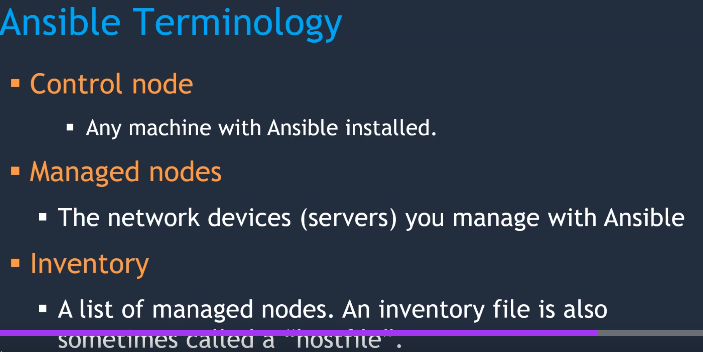
To manage nodes we need to enable password less authentication.

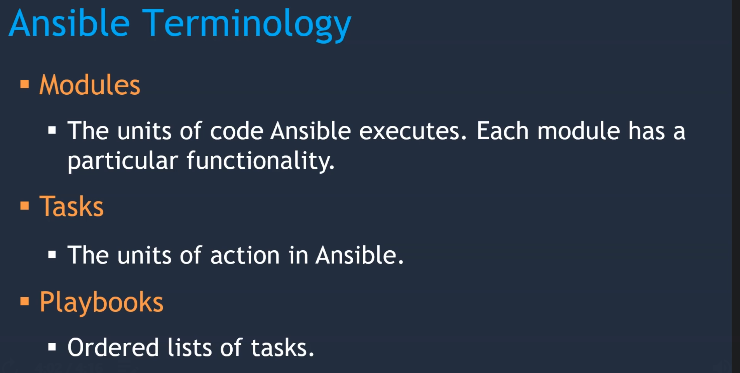
**Components in Ansible nodes:**

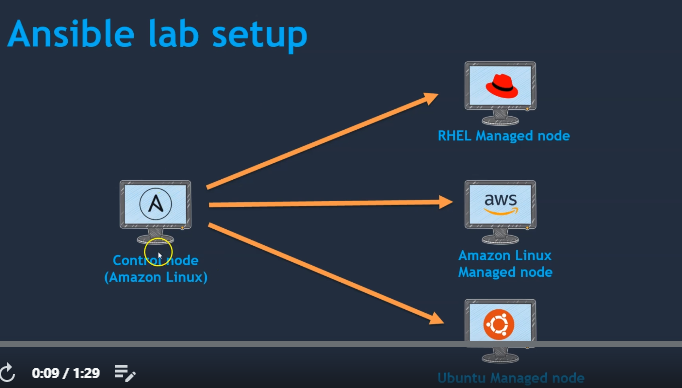
**Inventory:** This contains all the nodes managed by control node, it is also called as host file

**Playbooks:** Actual task which we need to execute on target systems.

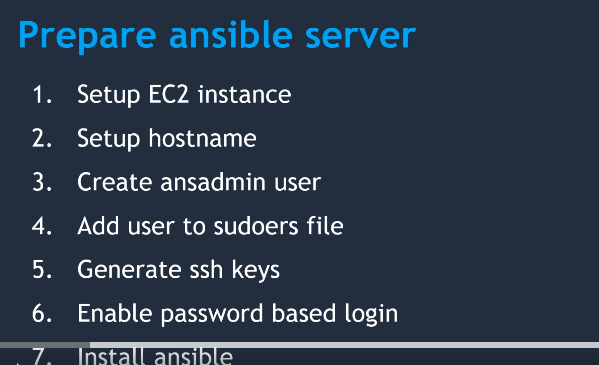
**Modules:** these will comes why installing Ansible and these modules used by playbooks.



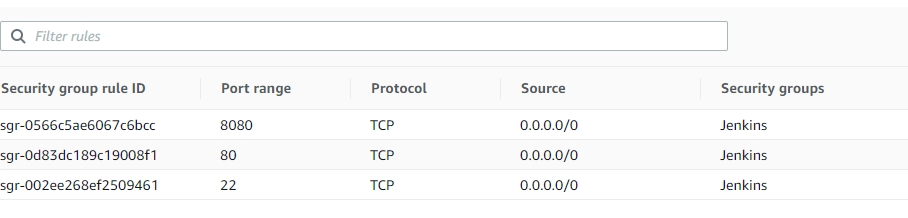




**Ansible cannot be installed on windows which means control node cannot be windows and it is always Linux and managed nodes can be windows which means by control node we can install or configure anything on windows node.**

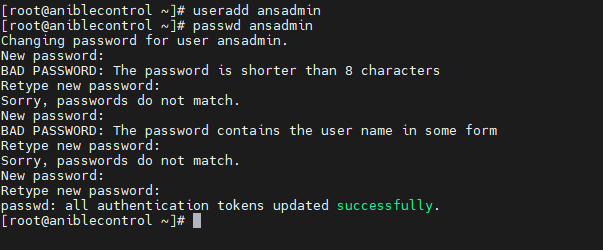


**While installing Ansible we need follow inbound rules in security group**

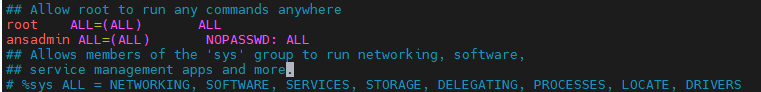


**Once instance is created then launch instance from mobaxterm using SSH,**

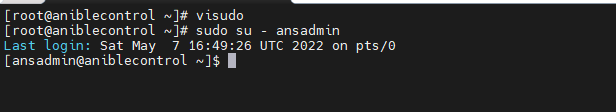
**Create one user by useradd command and password**



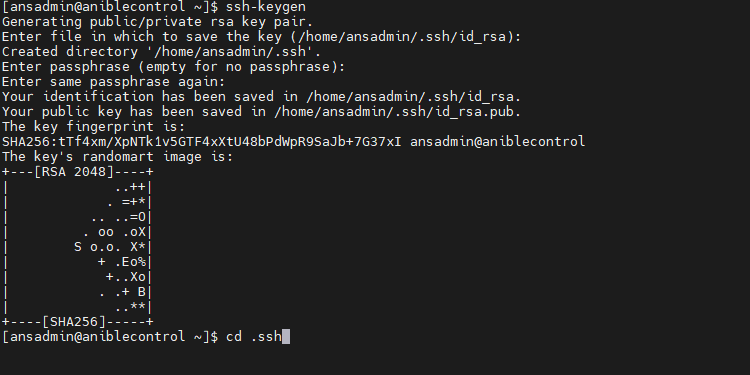
**Add users to sudoers file (**command is **visudo)**



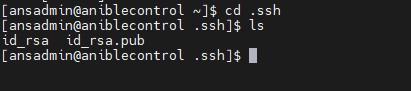
**Save and switch to created user**



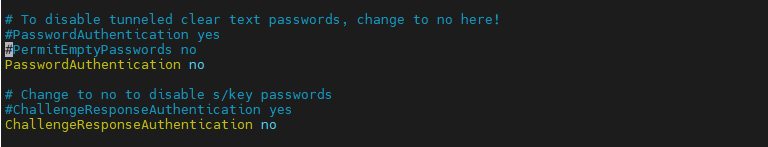
**Create SSH keys ( ssh-keygen is command)**



**Once created then cd into .ssh directory where we can see public key and private key, we need to copy this public key into targeted systems and with the help of private key we need to connect with public key**



**Now enable password based login, (vi /etc/ssh/sshd\_config)**

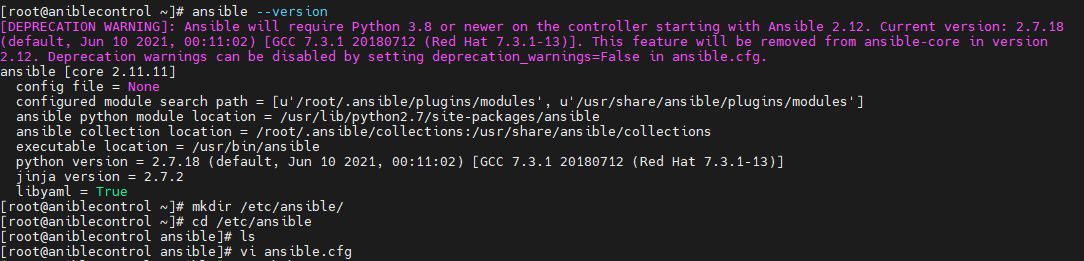


**Uncomment password authentication as Yes row and comment password authenticaton no row**

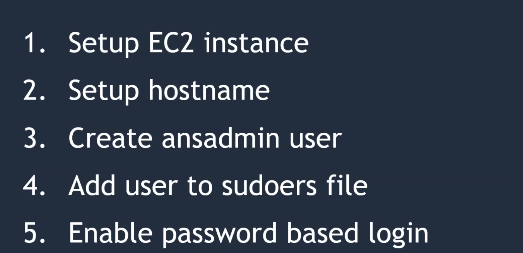


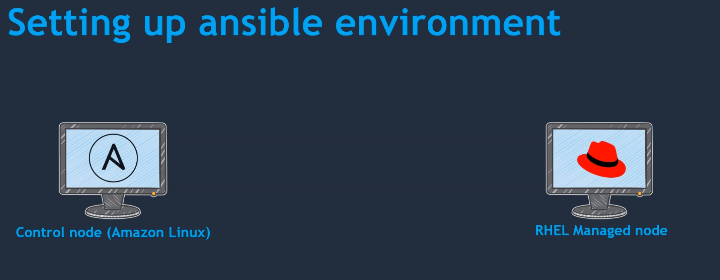
**Now install ansible**

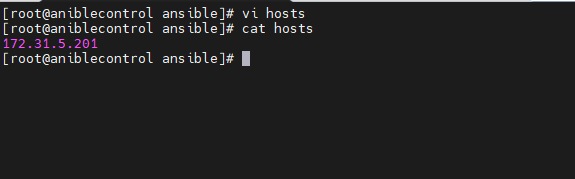
[**https://github.com/yankils/ansible/blob/master/install\_ansible\_on\_Amazon\_linux.md**](https://github.com/yankils/ansible/blob/master/install_ansible_on_Amazon_linux.md)

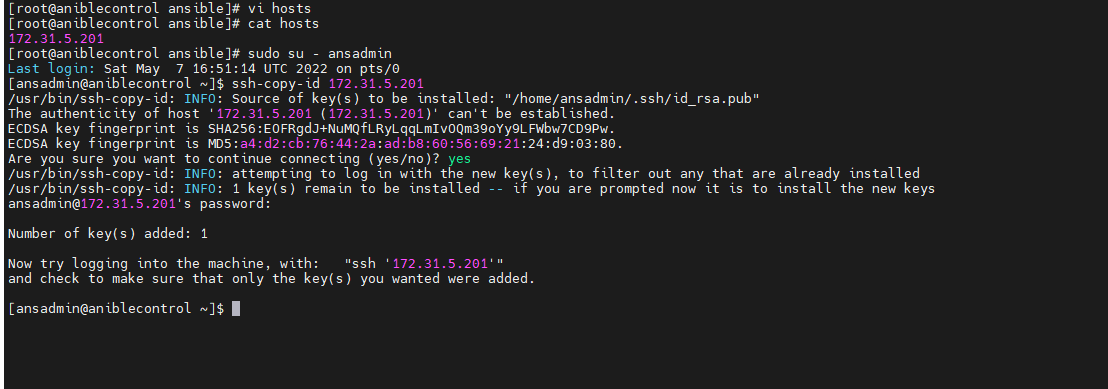


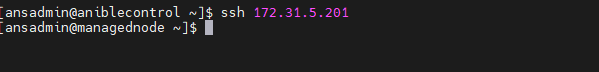
**We created ansible control node, we need to create managed node now. Steps for managed nodes**





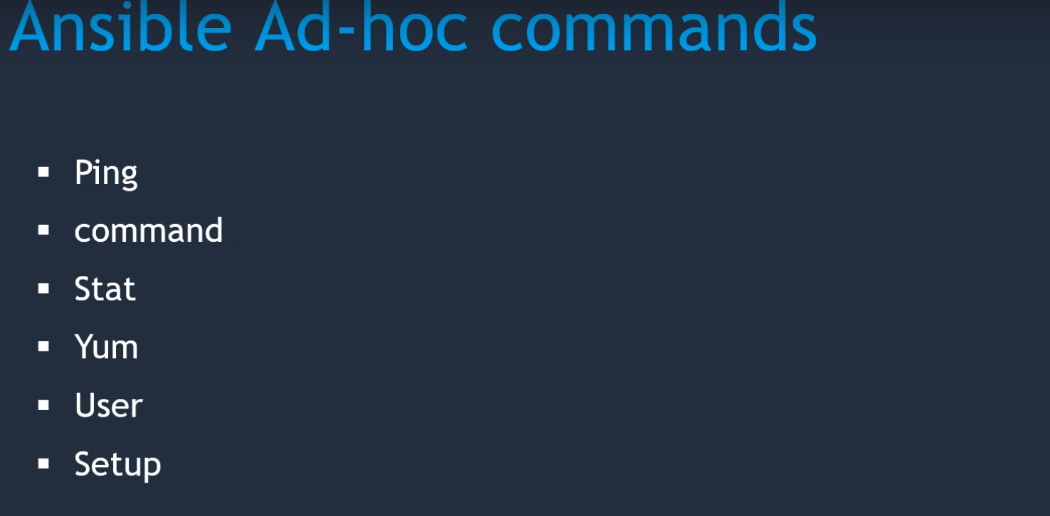






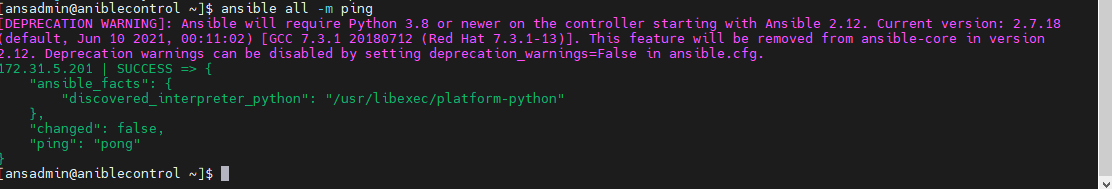
# Introduction to ad hoc commands[ℑ](https://docs.ansible.com/ansible/latest/user_guide/intro_adhoc.html#introduction-to-ad-hoc-commands)

An Ansible ad hoc command uses the /usr/bin/ansible command-line tool to automate a single task on one or more managed nodes. ad hoc commands are quick and easy, but they are not reusable. So why learn about ad hoc commands first? ad hoc commands demonstrate the simplicity and power of Ansible.

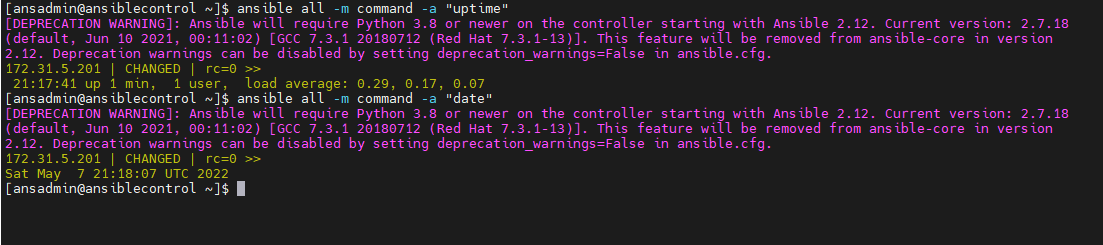


**We need to change root user to ansadmin used for ansible then use ping command**

**Ping module**



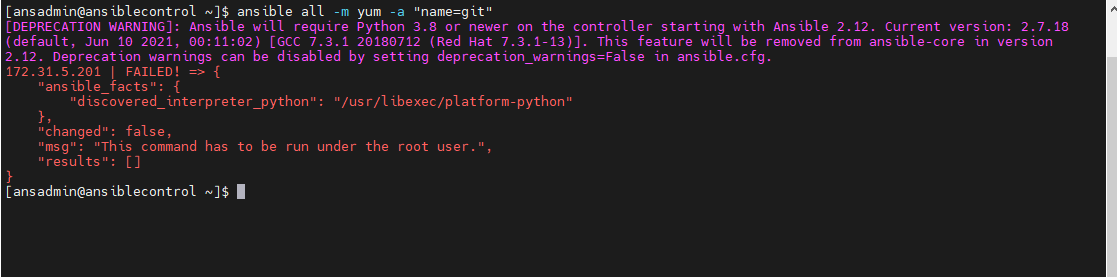
**Command module : ansilble all –m command –a “ “**



**Now in managed node we can see git is not installed so we can install it from control node by using yum module**

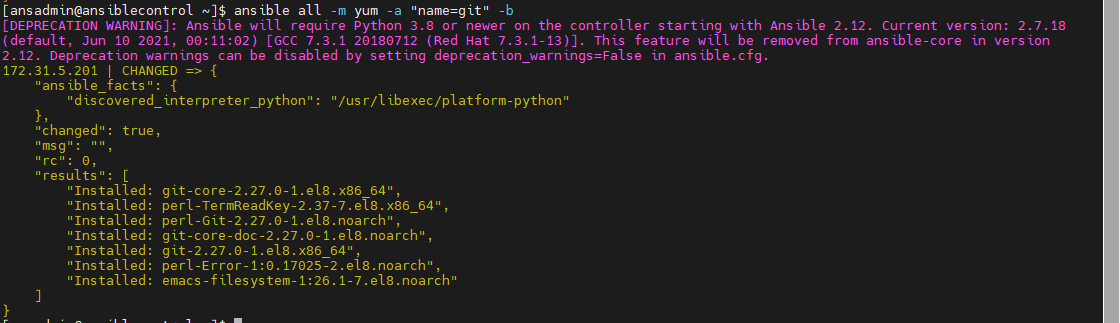


**Ansible all –m yum –a “name=git”**

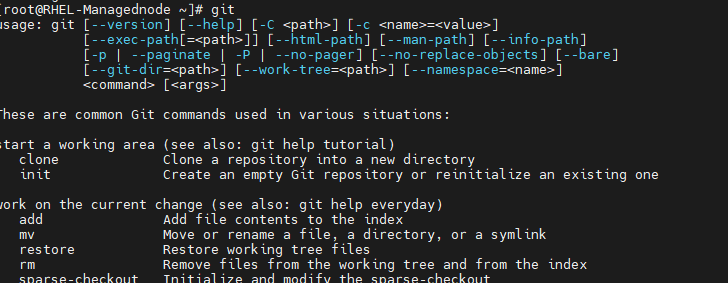


**As it mentioned we need to be root user so use become –b)**

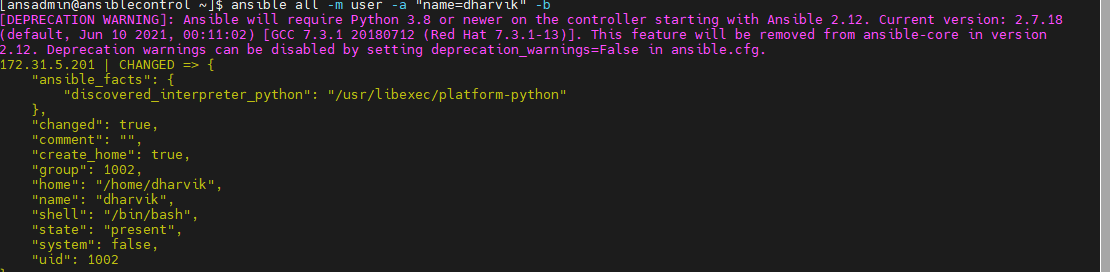
**Ansible all –m yum –a “name=git’ –b**



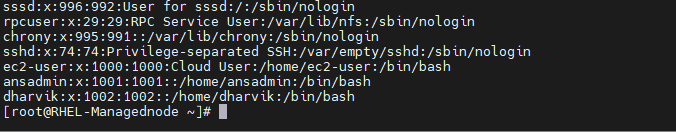
**Now we can see git on managed node**



With the user command we can create users on managed nodes

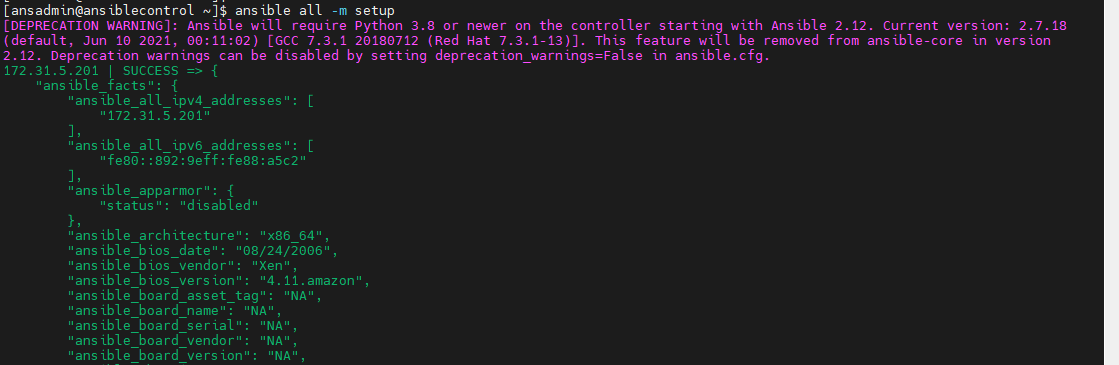


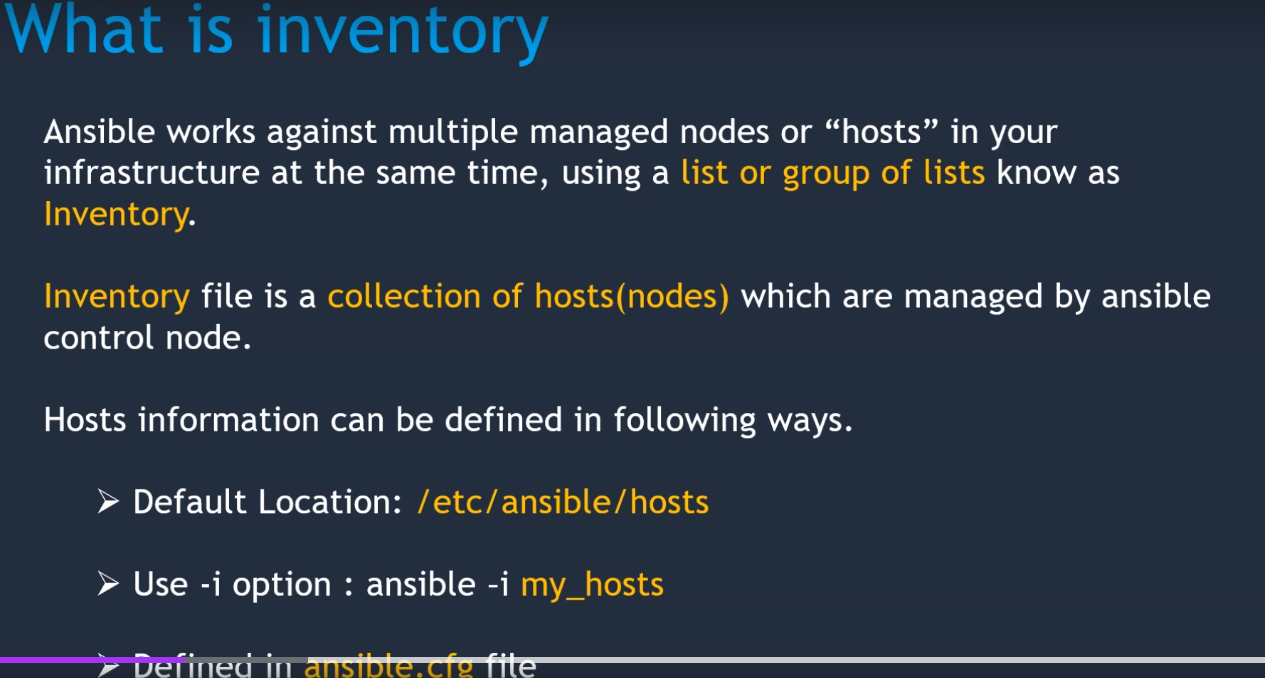
**User Dharvik is created on managed nodes**



**Setup command gives all the information (like network information, cpu information system information) about managed nodes**

**Ansible all –m setup**





**Ansible all –m ping –i hosts -- this syntax is for to use my own inventories.**

**If we need to change default inventory then we need to change it in ansible.cfg file, if multiple teammemebers running the ansible palybooks then using default is not suggestible and better to use inventories on their own.**

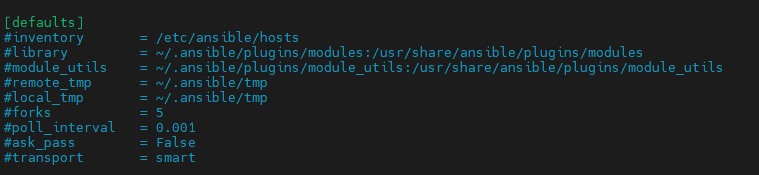
**We can create groups with in inventory and if required we can run specific groups only, for this we need to add group in syntax**

**For example in my inventory I have rhel and dummy groups and if I want to ping only rhel group then**

* **Ansible rhel –m ping – i /etc/ansible/hosts**

**-m this is to specifiy the module like –m ping which is module ping, if we don’t give –m then default module in configiuration file will be executed and for this uncomment default module row**

**Ansible configuration file: important details in configuration file**



**Forks : which tells maximum at a time maximum number of managed nodes**

**Ask\_pass ---- while login into client system it will ask password if it id true**

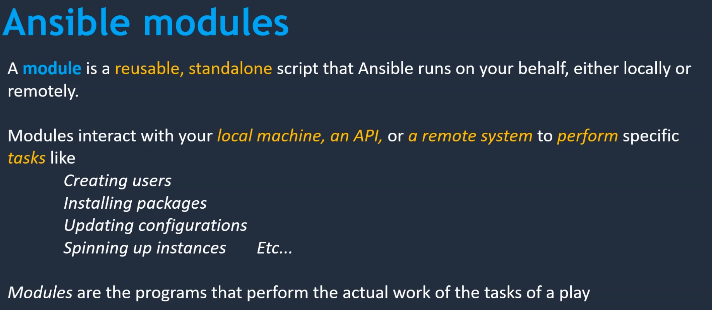
## The configuration file[ℑ](https://docs.ansible.com/ansible/latest/reference_appendices/config.html#the-configuration-file)

Changes can be made and used in a configuration file which will be searched for in the following order:

* ANSIBLE\_CONFIG (environment variable if set)
* ansible.cfg (in the current directory)
* ~/.ansible.cfg (in the home directory)
* /etc/ansible/ansible.cfg

Ansible will process the above list and use the first file found, all others are ignored.

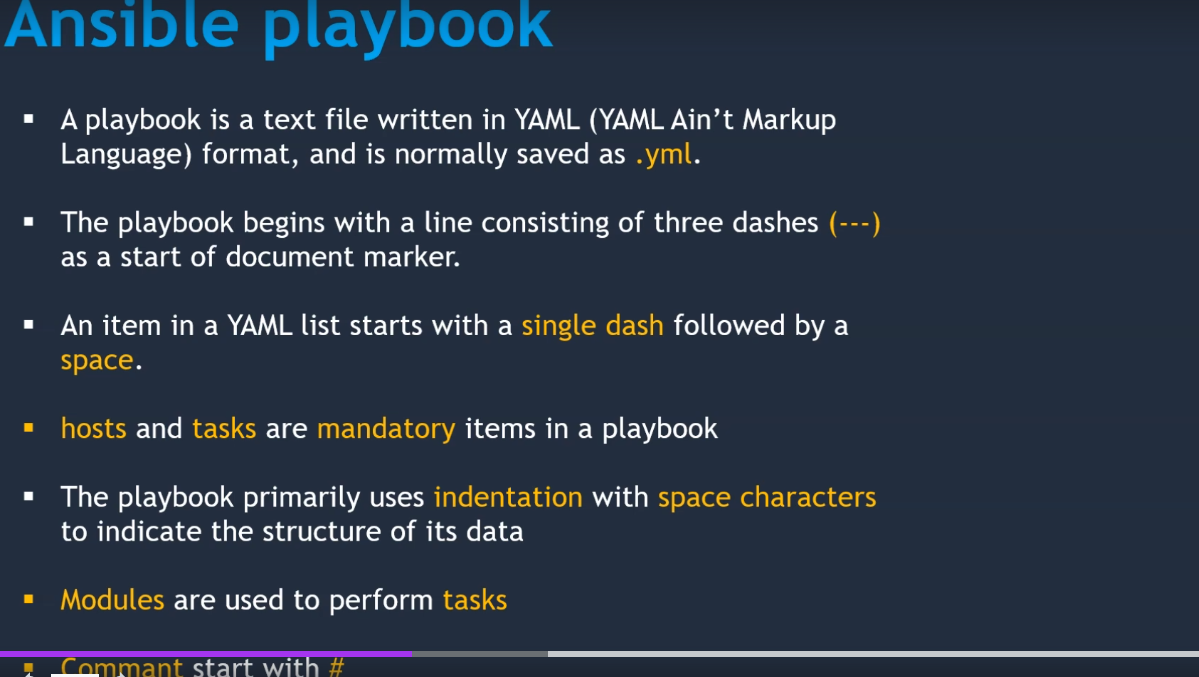
**Modules:**

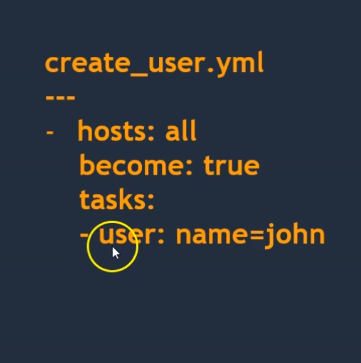


[**https://docs.ansible.com/ansible/2.8/user\_guide/modules.html**](https://docs.ansible.com/ansible/2.8/user_guide/modules.html)

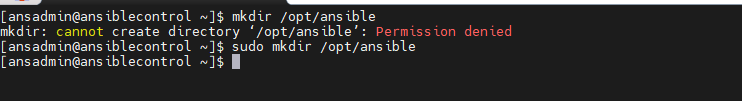
**Ansible Playbooks:**

Playbooks are set of instructions which can be executed on group of target systems or single system.

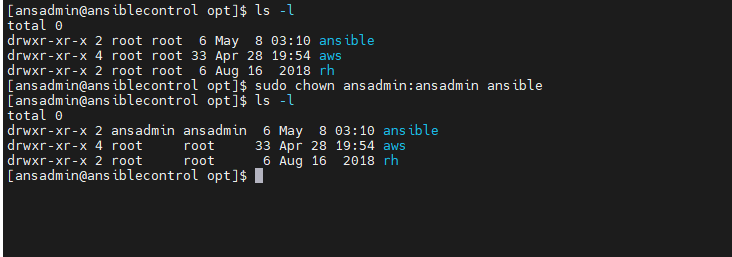




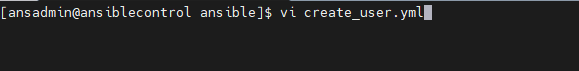
**Create one new folder ansible in opt**



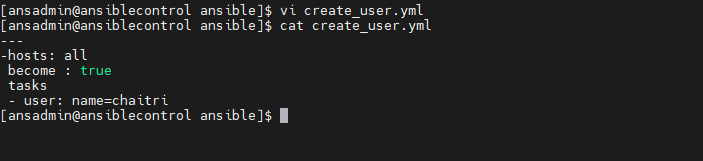
**Change ownership**

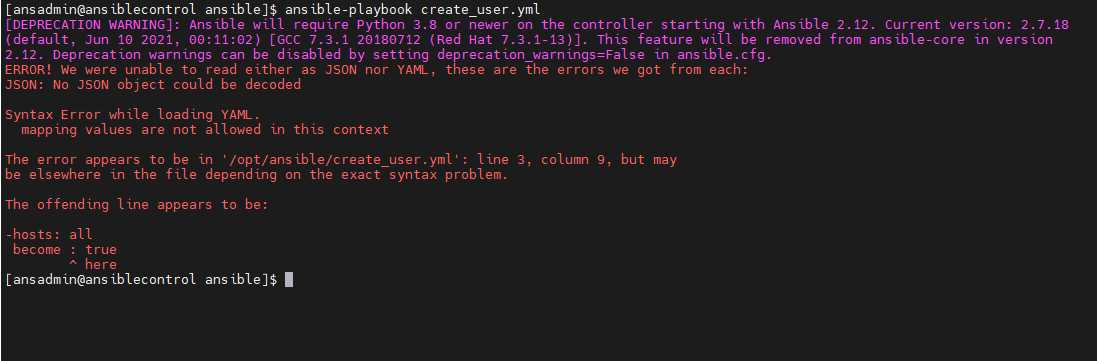


**Now create one file in ansible**

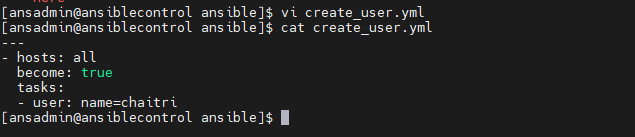


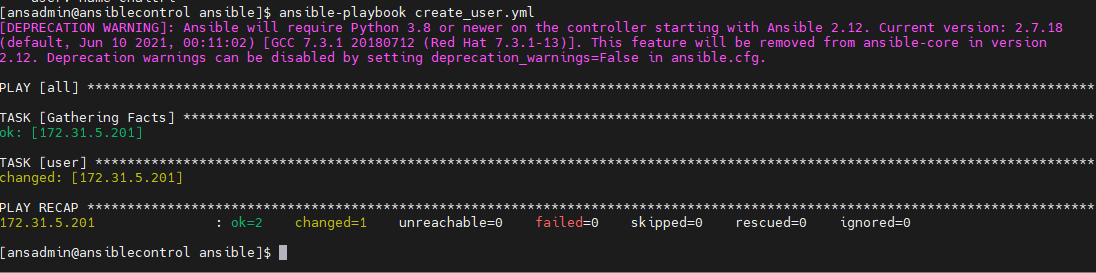




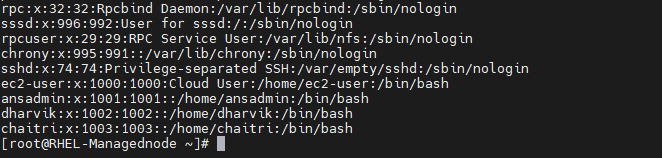


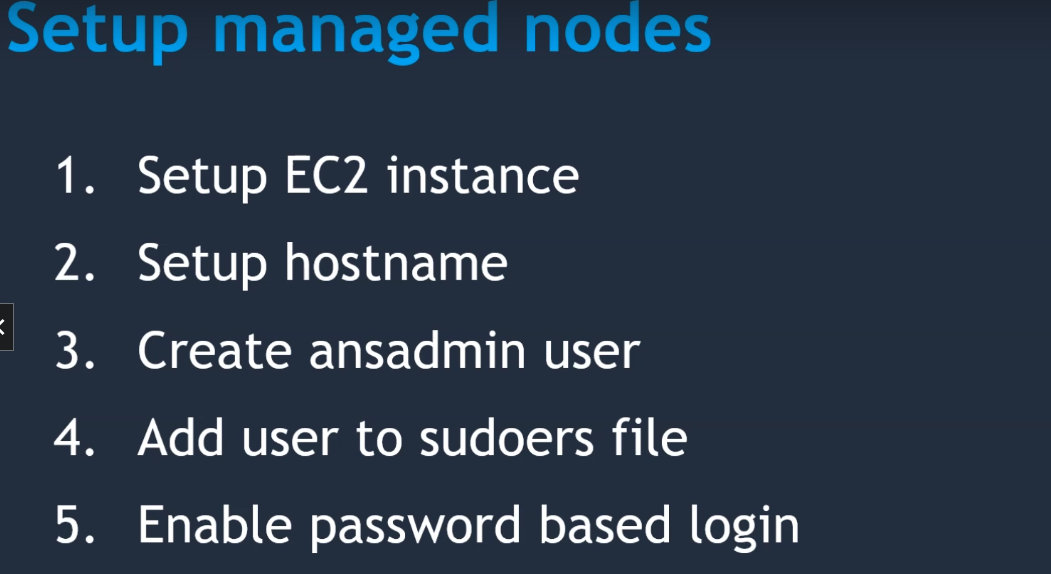
**Changed spaces**



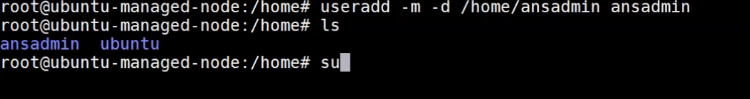


**Now we can see new user chaitri in managed node**

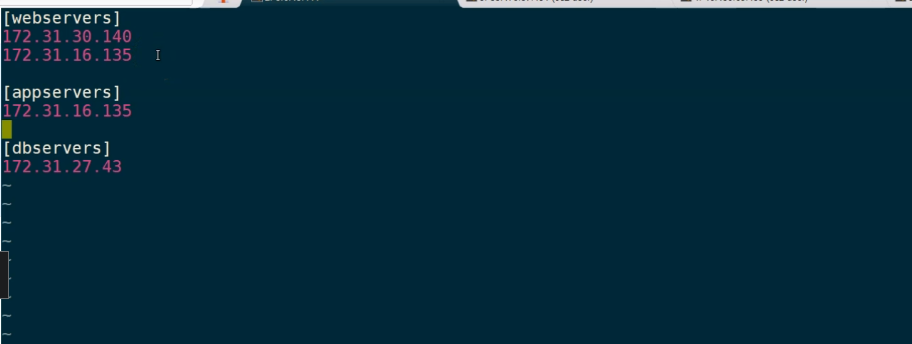


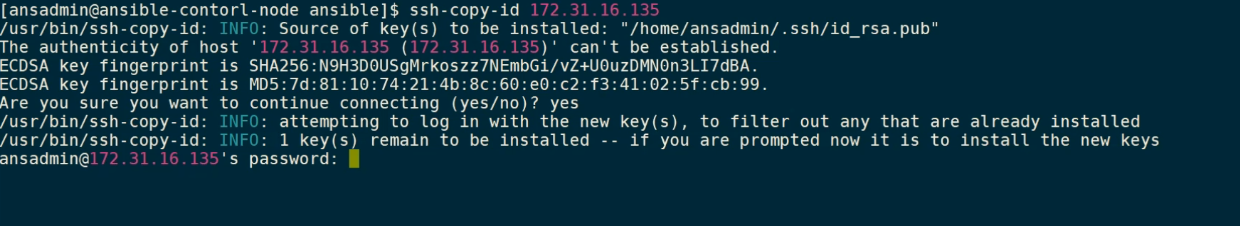


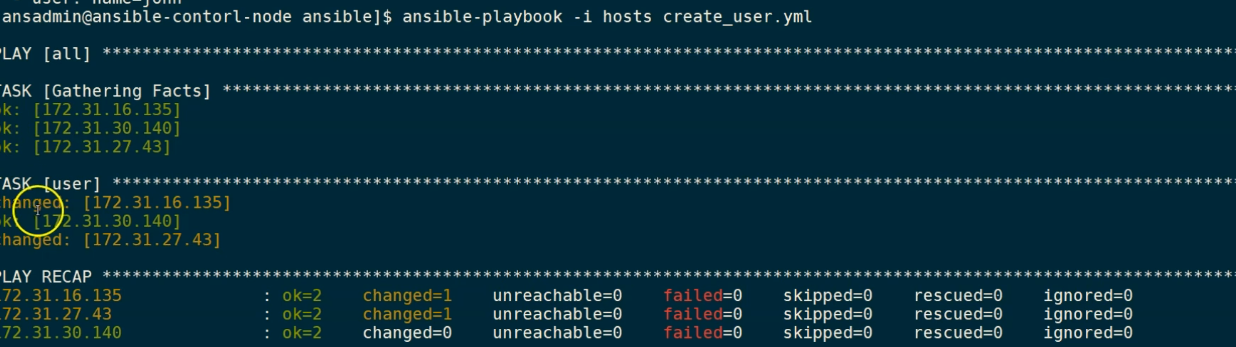
**Creating user in ubuntu**



**Hosts file**







Gathering facts is the default modules run on every playbook

### 1. What is Configuration Management?

It’s a practice that we should follow in order to keep track of all updates that are going into the system over a period of time. This also helps in a situation where a major bug has been introduced to the system due to some new changes and we need to fix it with minimum downtime. Instead of fixing the bug, we can roll back the new changes(which caused this bug) as we have been tracking those

### 3. How does Ansible work?

Ansible is a combination of multiple pieces working together to become an automation tool. Mainly these are modules, playbooks, and plugins.

* Modules are small codes that will get executed. There are multiple inbuilt modules that serve as a starting point for building tasks.
* Playbooks contain plays which further is a group of tasks. This is the place to define the workflow or the steps needed to complete a process
* Plugins are special kinds of modules that run on the main control machine for logging purposes. There are other types of plugins also.

The playbooks ran via an Ansible automation engine. These playbooks contain modules that are basically actions that run in host machines. The mechanism is followed here is the push mechanism, so ansible pushes small programs to these host machines which are written to be resource models of the desired state of the system.

### 4. What are the features of Ansible?

It has the following features:

* **Agentless** – Unlike puppet or chef there is no software or agent managing the nodes.
* **Python** – Built on top of python which is very easy to learn and write scripts and one of the robust programming languages.
* **SSH**– Passwordless network authentication which makes it more secure and easy to set up.
* **Push architecture** – The core concept is to push multiple small codes to the configure and run the action on client nodes.
* **Setup** – This is very easy to set up with a very low learning curve and any open source so that anyone can get hands-on.
* **Manage Inventory** – Machines’ addresses are stored in a simple text format and we can add different sources of truth to pull the list using plugins such as Openstack, Rackspace, etc

### 5. What is a YAML file and how do we use it in Ansible?

YAML or files are like any formatted text file with few sets of rules just like JSON or XML. Ansible uses this syntax for playbooks as it is more readable than other formats.

### 20. What is the difference between Ansible and Puppet?

**Management and Scheduling:**  In Ansible, the server pushes the configuration to the nodes on the other hand in puppet, the client pulls the configuration from the server. Also for scheduling, the puppet has an agent who polls every 30mins(default settings) to make sure all nodes are in a desirable state. Ansible doesn’t have that feature in the free version.  
**Availability:** Ansible has backup secondary nodes and puppet has more than one master node. So both try to be highly available.  
**Setup:** Puppet is considered to be harder to set up than ansible as it has a client-server architecture and also there’s a specific language called Puppet DSL which is its own declarative language