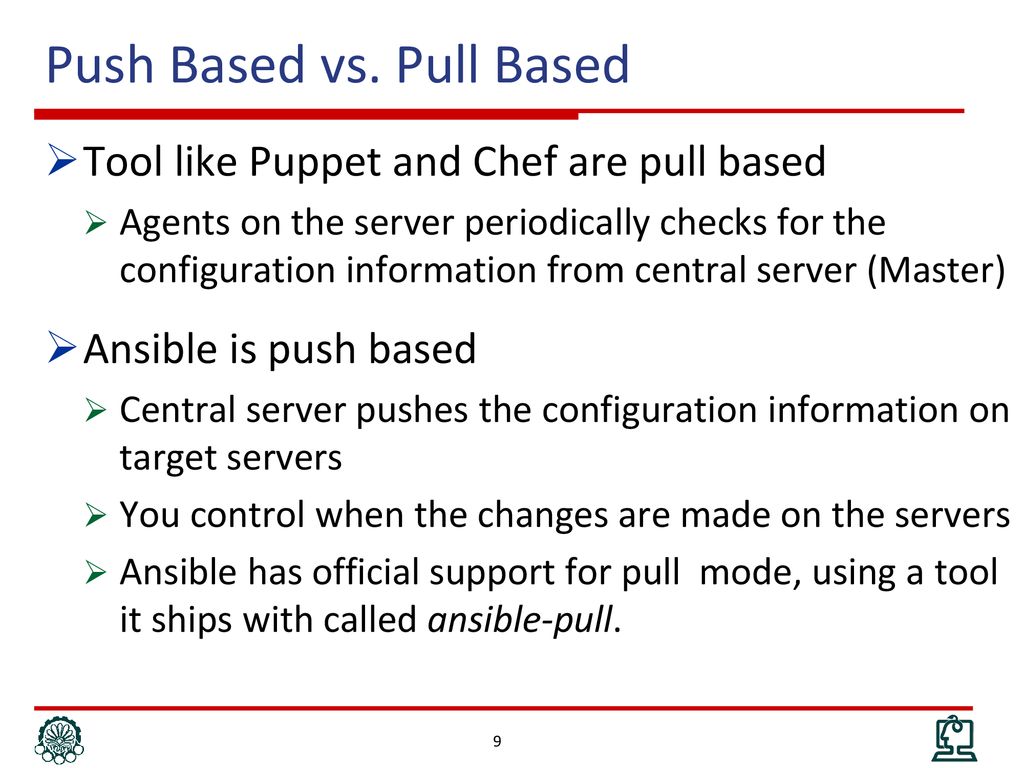
Ansible is an open-source software provisioning, configuration management, and application-deployment tool enabling infrastructure as code

Ansible is easy to deploy because it does not use any agents or custom security infrastructure.



P—Playbook,P—plays,T—tasks,M—modules

(Playbook contains many plays,a play contains many tasks ,a tasks contains many modules or roles)

**Installation**

Open source—100 nodes

Enterprise---more than 100 nodes,,(Ansible tower)

\*No need to install anything on clients

\*parallel(By default)

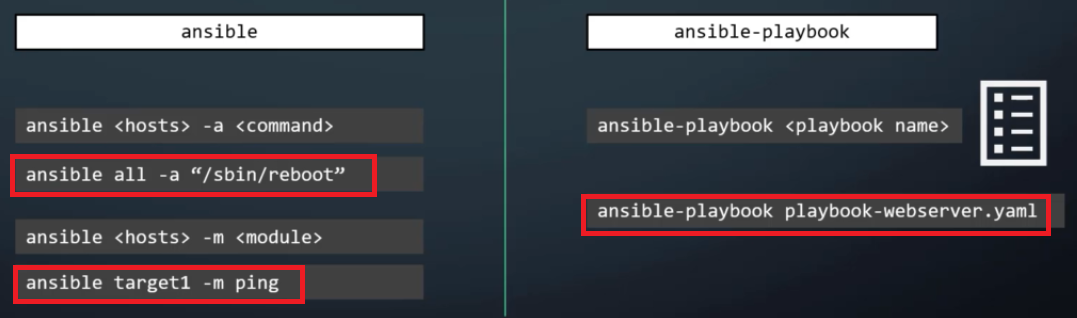
\*Serial

\*-vvv

\*\*Tasks must be ordered properly

\*include—to include another playbook





**Ansible terminology**

**Inventory**: It is a file contains a list of target hosts, groups on which need to perform the tasks through the playbook.

Default ansible inventory **-----/etc/ansible/hosts**

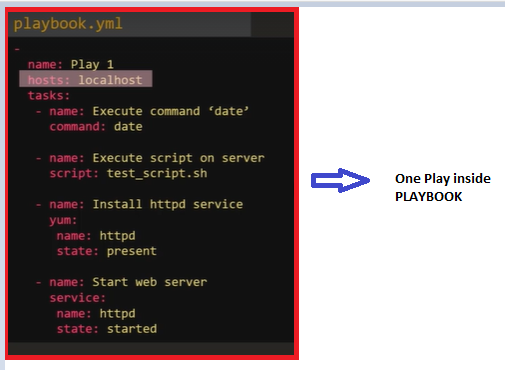
**Facts**: Data gathered from target hosts,when we execute a playbook the ansible facts collects the info about the hosts **(setup module)**

**Playbook:** An Ansible playbook contains one or more plays, each of which tell Ansible master what to execute/perform on target machines.

**Play**: Defines set of tasks to be executed on hosts or targets.

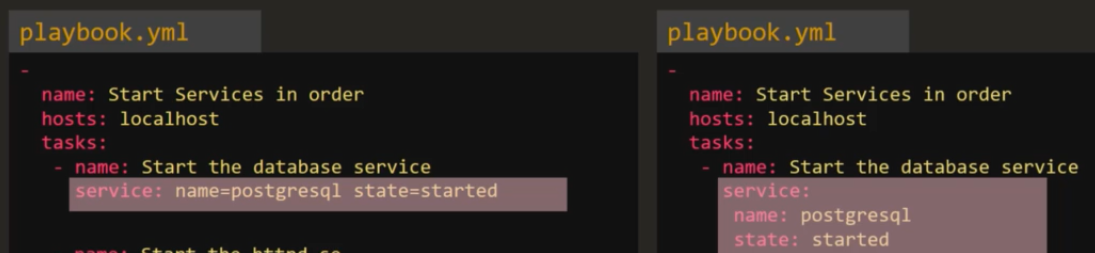
**Tasks**: It includes one or more modules, to perform the action on hosts based on the module.

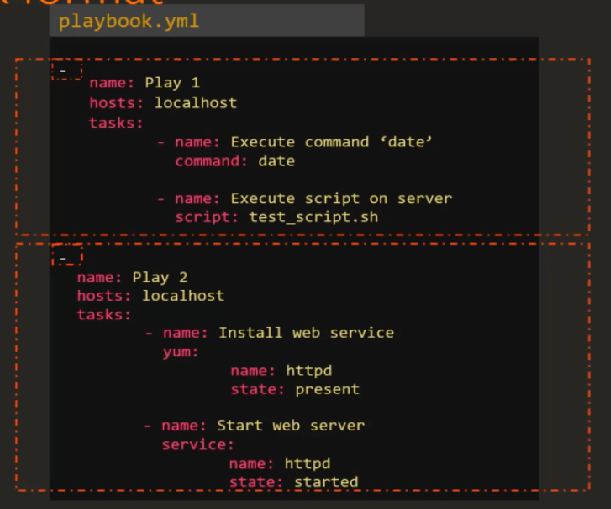
**Modules :** The diff actions run by tasks on target machines are called modules.



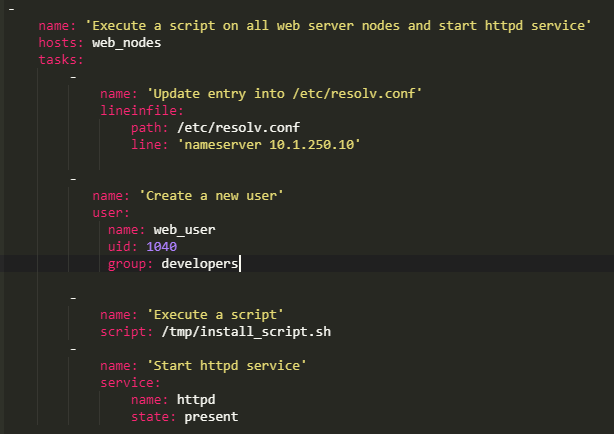
1)Command 2)Script 3)service 4) lineinfile 5)user 6)Setup 8)shell 9)yum,apt 10)template11) raw(redirect) 12)Expect—interactive execution 13)Debug

14)include

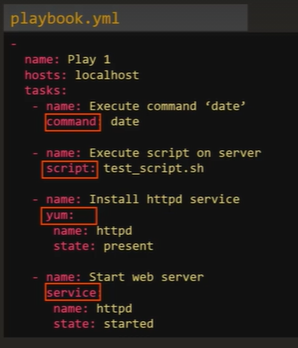




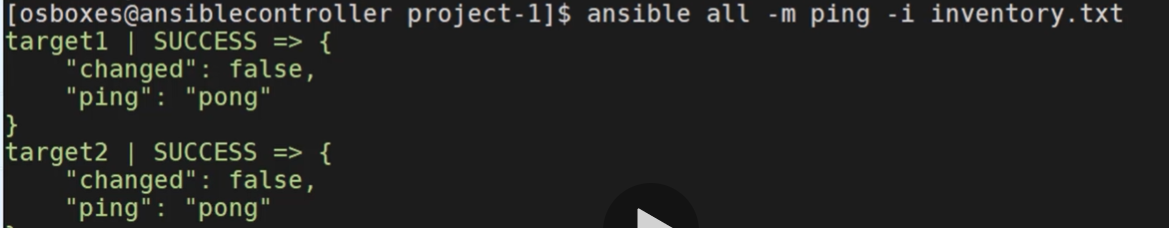




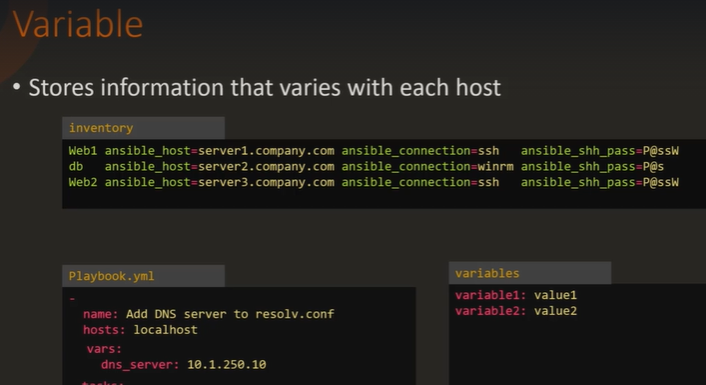


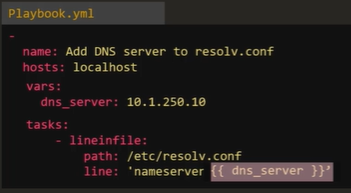


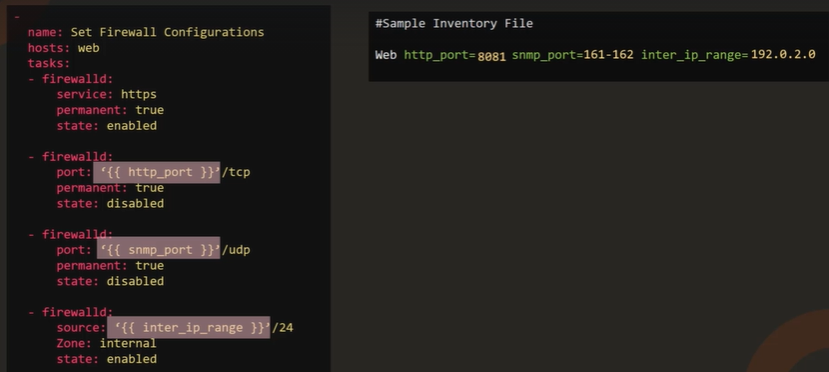
**Imparative: way**

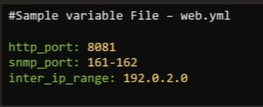


**Variables:** Variables are used to store values that varies



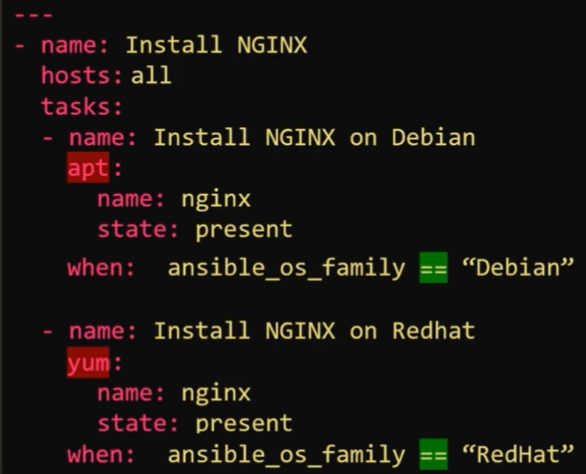


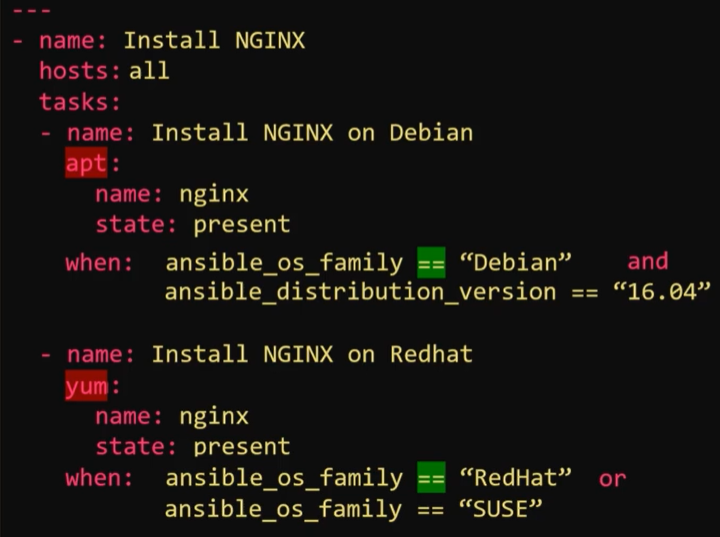




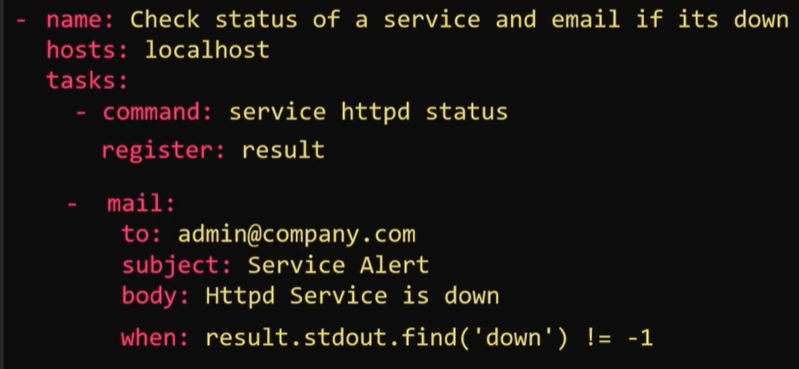
**Jinja2 template** :: Jinja2 templates are simple template files that store variables that can change from time to time

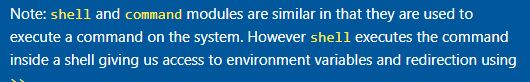
**Conditional:**



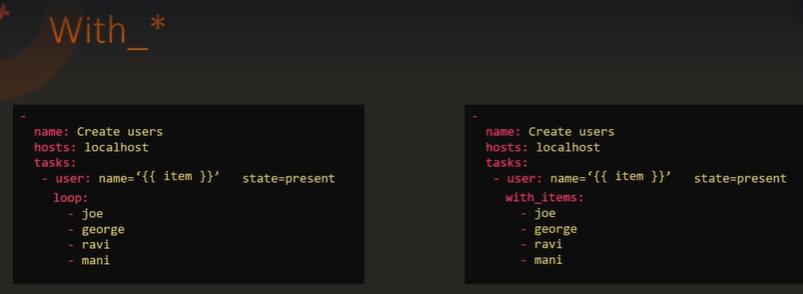


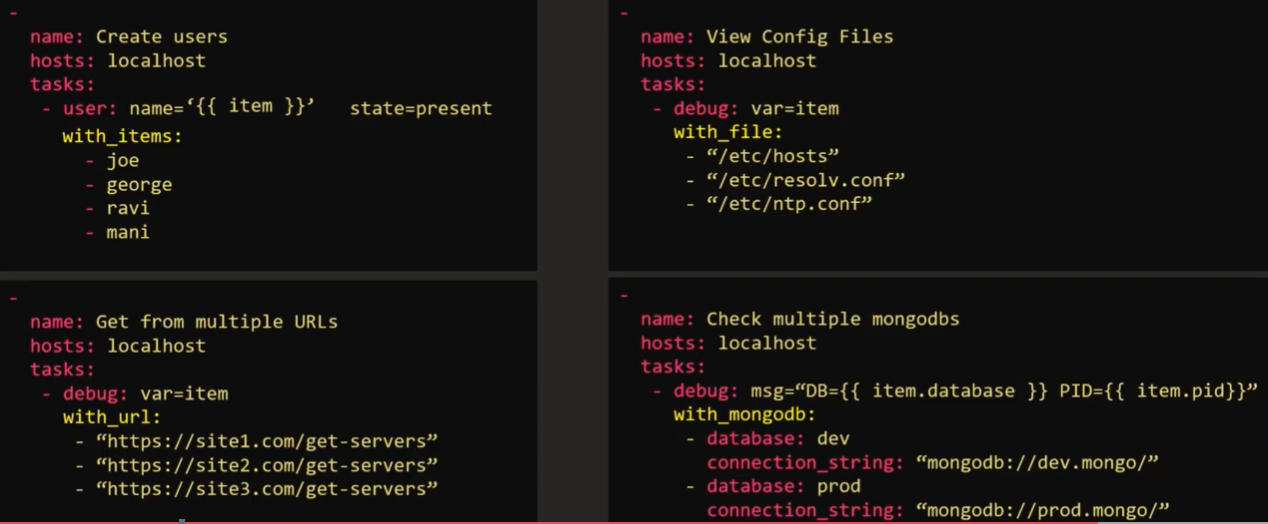
**Register ::** To record the output of one task we can use register directive





**Loops:**



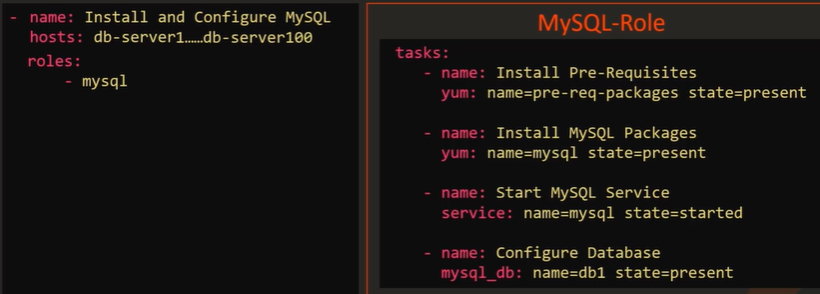


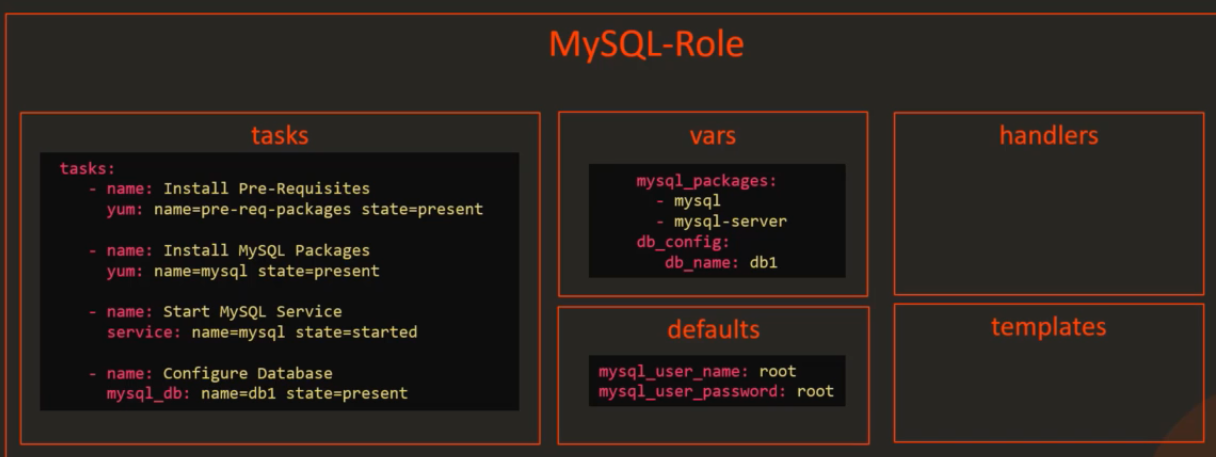
**Role:**

\*Roles make it easy to develop,reuse and share ansible playbooks

Roles provide a framework for fully independent, or interdependent collections of variables, tasks, files, templates, and modules.

In Ansible, the role is the primary mechanism for breaking a playbook into multiple files

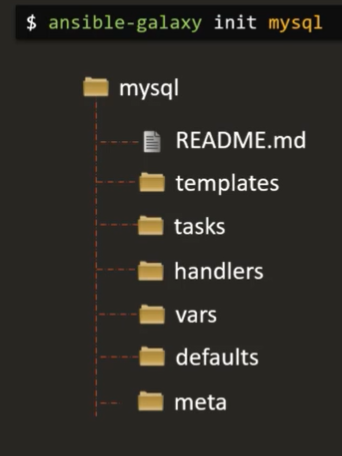




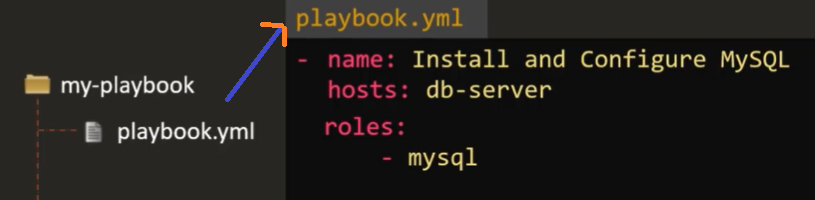
**Galaxy** :: Ansible Galaxy is a repository/site for Ansible Roles that are available to drop directly into your Playbooks to streamline your automation projects..We can download ,share custom roles.

**Role needs directory structure ,ansible galaxy as native tool that creates a scalaton**

**To create role**

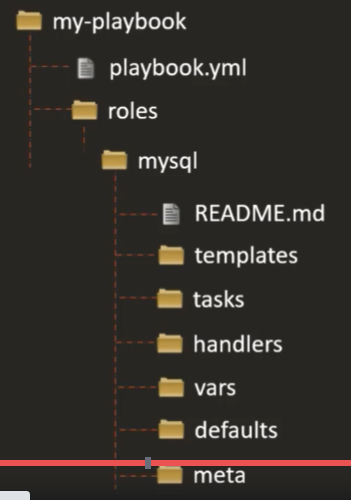


**How to use roles :**



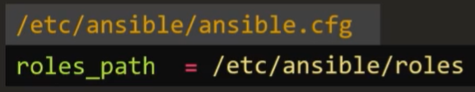
**How playbook finds where the role is**

1. **Create a directory called “role” under within playbook folder.and move the roles on to it.**

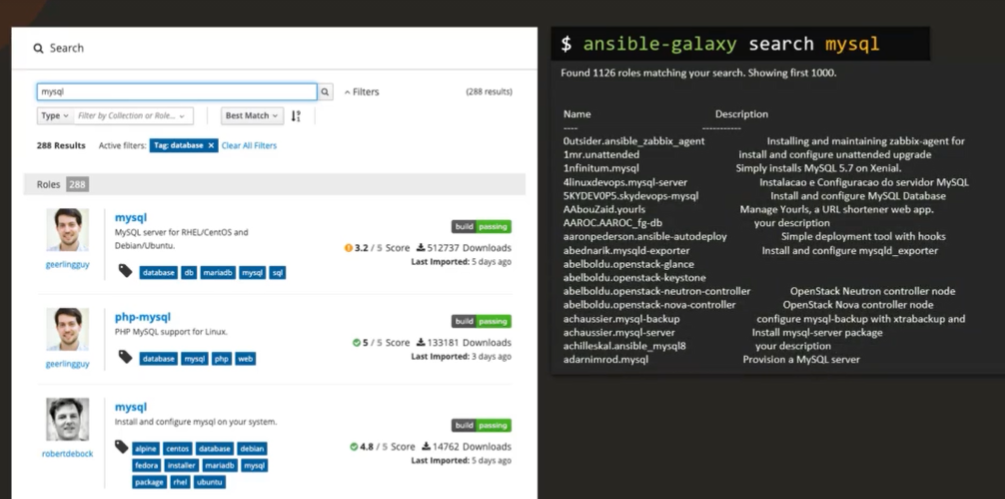


1. **U can move the roles to a common directory dedicated for roles on the system “etc/ansible/roles” it’s the default location where ansible searches for roles**

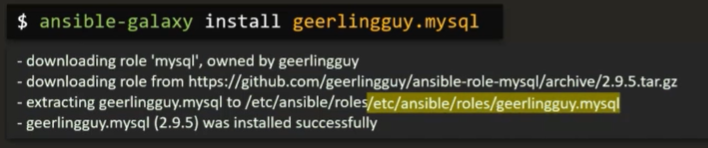
**This is path is mentioned in the ansible config file.**

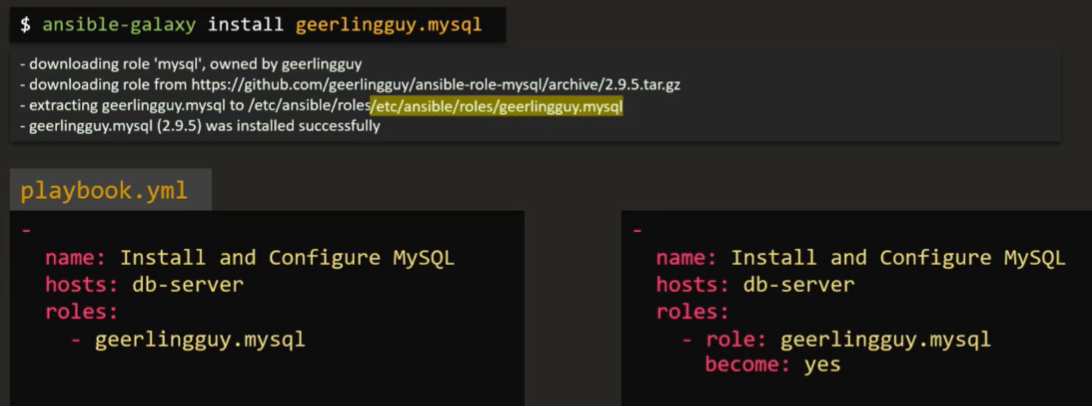


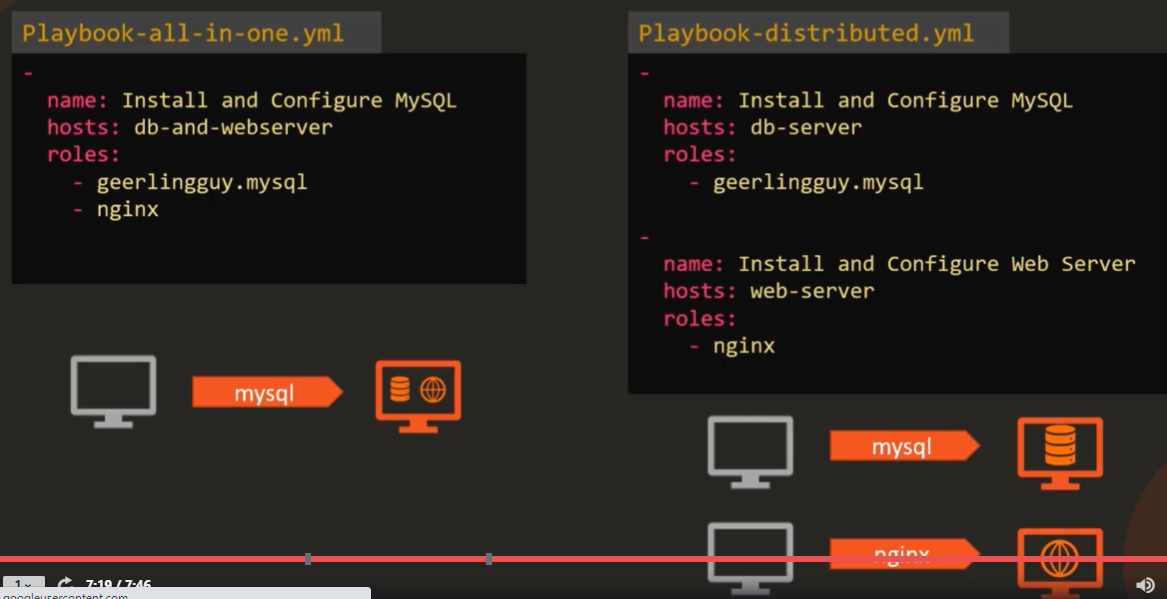
**How to find roles:  
1) on the galaxy UI or from the cmd (ansible galaxy search)**

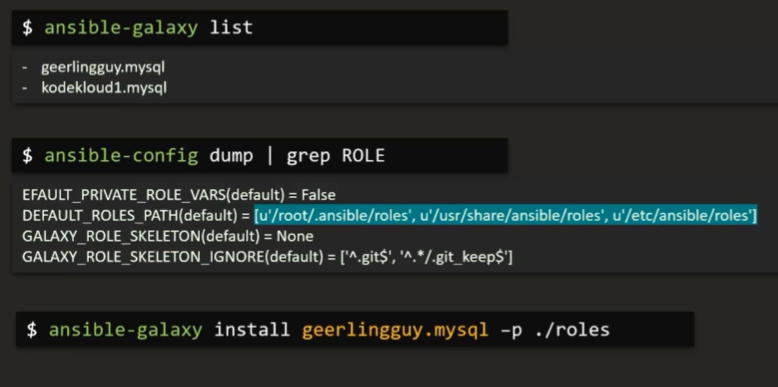


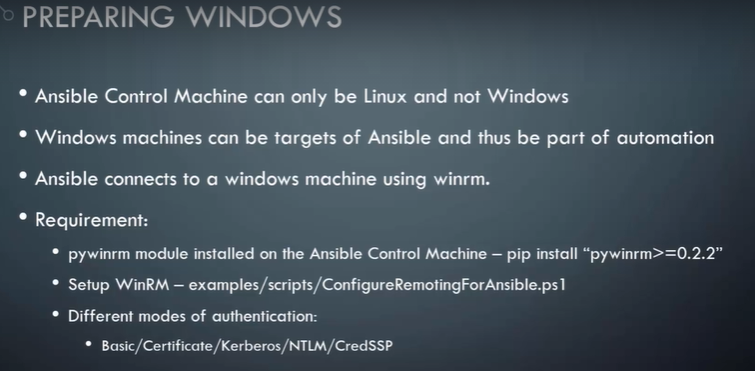
**How to use roles(role will be downloaded under the directory /etc/ansible/roles/nameof the role)**

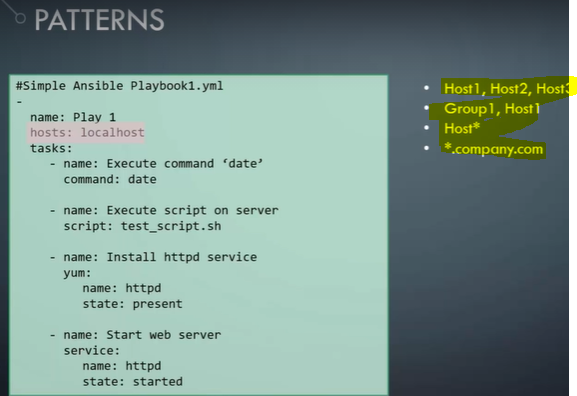












## **Types of Ansible Inventories:**

* **Static Inventory**
* **Dynamic Inventory**

## **Static Inventory**

Static inventory is default inventory and is defined in the /etc/ansible/ansible.cfg file.

Default file can be changed inside the ansible.cfg file.

If you want to use the custom file as inventory input can specify it using ” -i  /path/to/file ” with Ansible command line.

**Example of Static Inventory:**

The below example Ansible Ad-Hoc command will check the ping status of all servers part of group called “all”

**Example**

|  |  |
| --- | --- |
| 1 | ansible -i /etc/Ansible/hosts.ini -m ping all |

## **Dynamic Inventory**

If you have the setup where you add and remove the hosts very frequently, then keeping your inventory always up-to-date become a little bit problematic. In such case Dynamic inventory comes into picture, generally are scripts (Python/Shell) for dynamic environments (for example cloud environments)

 With Ansible, as aforementioned, can use “-i” to specify the custom inventory file

**Dynamic inventory got benefits over static inventories:**

* Reduces human error, as information is collected by scripts.
* Very less manual efforts for managing the inventories.

**Ansible has inventory collection scripts for the below platforms**

* AWS EC2 External Inventory Script, Collber, OpenStack, BSD Jails, Google Compute Engine, and Spacewalk.

**Information Source:**[**http://docs.ansible.com/ansible/intro\_dynamic\_inventory.html**](https://docs.ansible.com/ansible/intro_dynamic_inventory.html)

For the quick demo, I have downloaded the Ansible Dynamic inventory script and related ec2.ini file from Ansible website.

dynamic-inven-ec.py             –        EC2 external inventory script (Python Script)

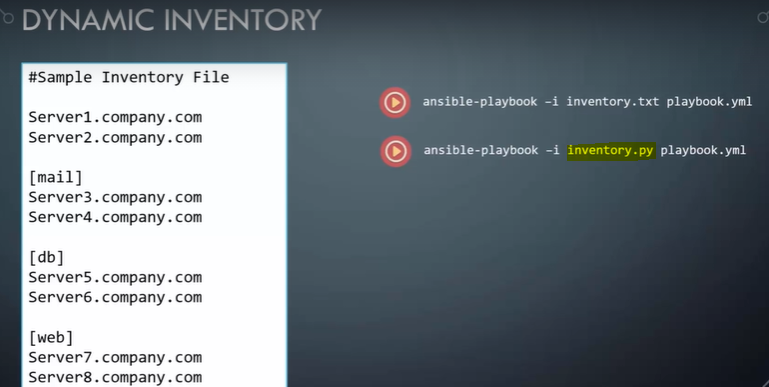
ec2.ini                                     –        Contains Ansible EC2 external inventory script settings

In the dynamic inventories, grouping can be managed by “tags, regions etc inside the ec2.ini file.

**Example:**

|  |  |
| --- | --- |
| 1 | # ansible -i dynamic-inven-ec.py   -u ec2-user tag\_Name\_awslab\*  -m ping |

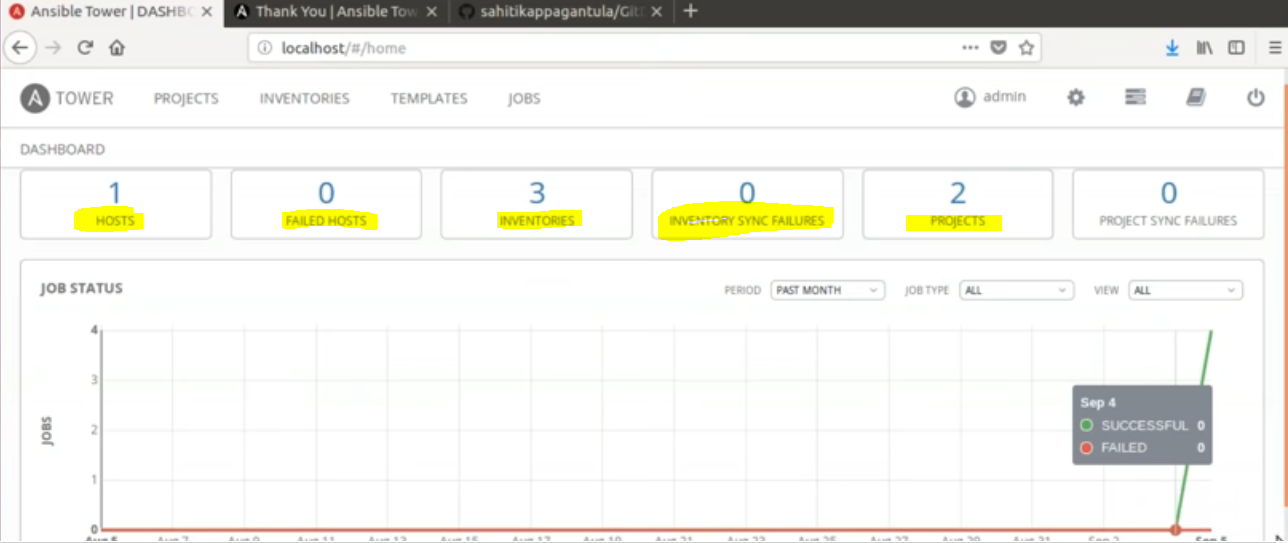
In the above example, Ansible will do the ping test for servers collected by dynamic inventory with tag key Name and Value with suffix “awslab”.

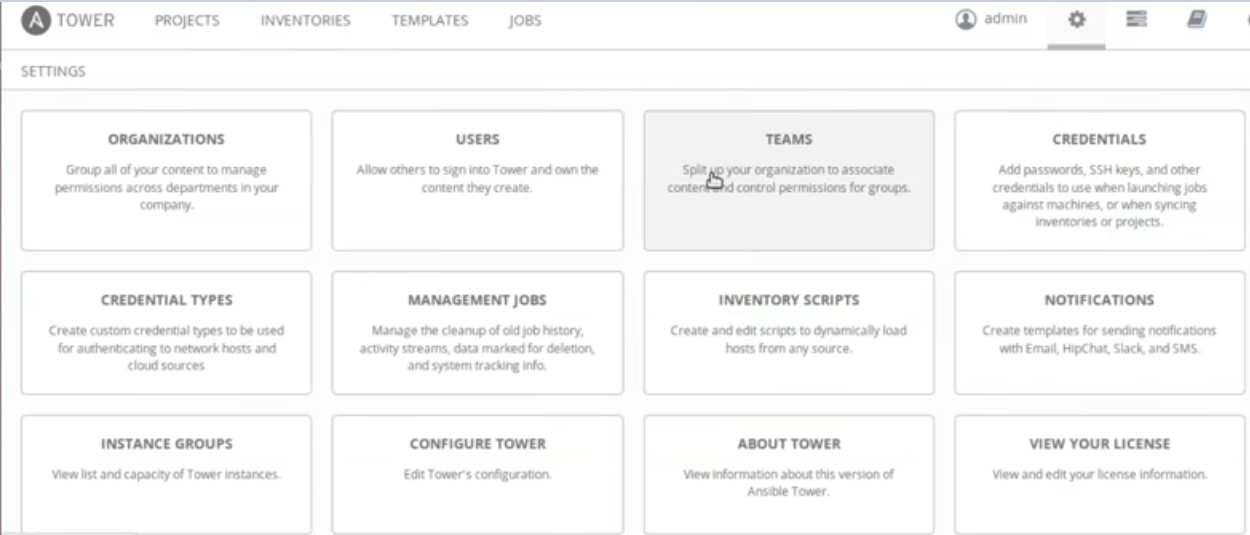


## **What is Ansible Tower?**

Ansible Tower is Ansible at a more enterprise level. It is a web-based solution for managing your organization with a very easy user interface that provides a dashboard with all of the state summaries of all the hosts, allows quick deployments, and monitors all configurations.

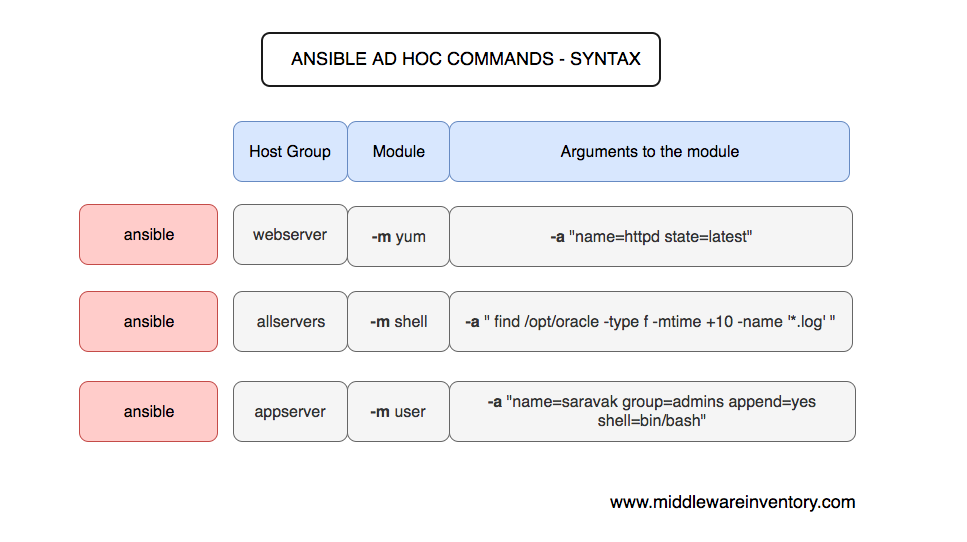
The tower allows you to share the SSH credentials without exposing them, logs all the jobs, manage inventories graphically and syncs them with a wide variety of cloud providers.



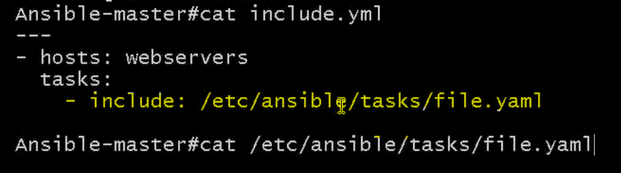


**AdHoc Commands**

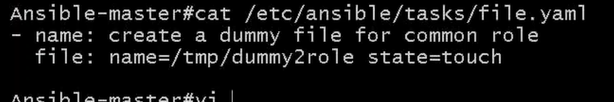
Adhoc commands are simple one line command to perform some action. Running modules with Ansible commands are adhoc commands.

[](https://www.middlewareinventory.com/wp-content/uploads/2019/01/Screen-Shot-2019-01-16-at-4.48.32-AM.png)

**Include module::** We can call another yml file inside one playbook.

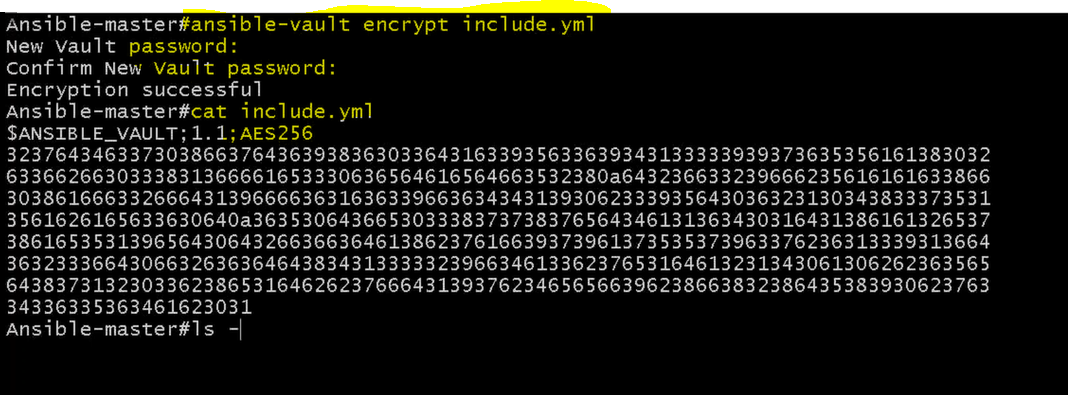


**This yml file does some tasks like installing any package ,creating a file,starting a service**



**Ansible vault :** Used to encrypts the playbook with a password,while executing the playbook ,it decrypts and executes.

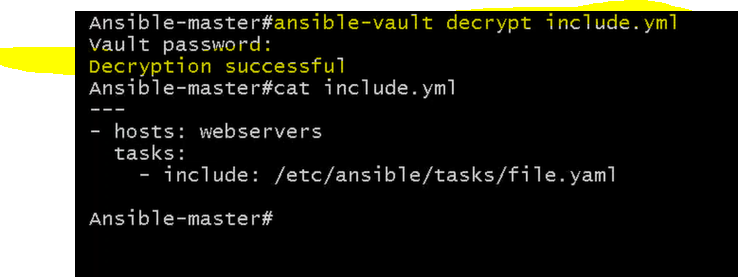
It is basically for security ,whenever we have some confidential information in the playbook we can restrict people from accessing it using ansible vault.



**While executing we don’t need to decrypt ,ansible will take care of that provided password is correct.**



**Decrypt:**



# Handlers:

Sometimes you want a task to run only when a change is made on a machine.

For example, you may want to restart a service if a task updates the configuration of that service, but not if the configuration is unchanged.

Ansible uses handlers to address this use case.

Handlers are tasks that only run when notified. Each handler should have a globally unique name.

