# ANSIBLE

- Ansible is an open-source configuration management tool
- Used for configuration management
- Can solve wide range of automation challenges
- Written by Michael  DeHaan
- In 2015 red hat acquired Ansible

## Advantage of Ansible

- Easy to learn
- Written in python
- Easy installation and configuration steps
- No need to install  ansible on slave
- Highly scaable

## Popularity of Ansible

Apple


NASA


Intel

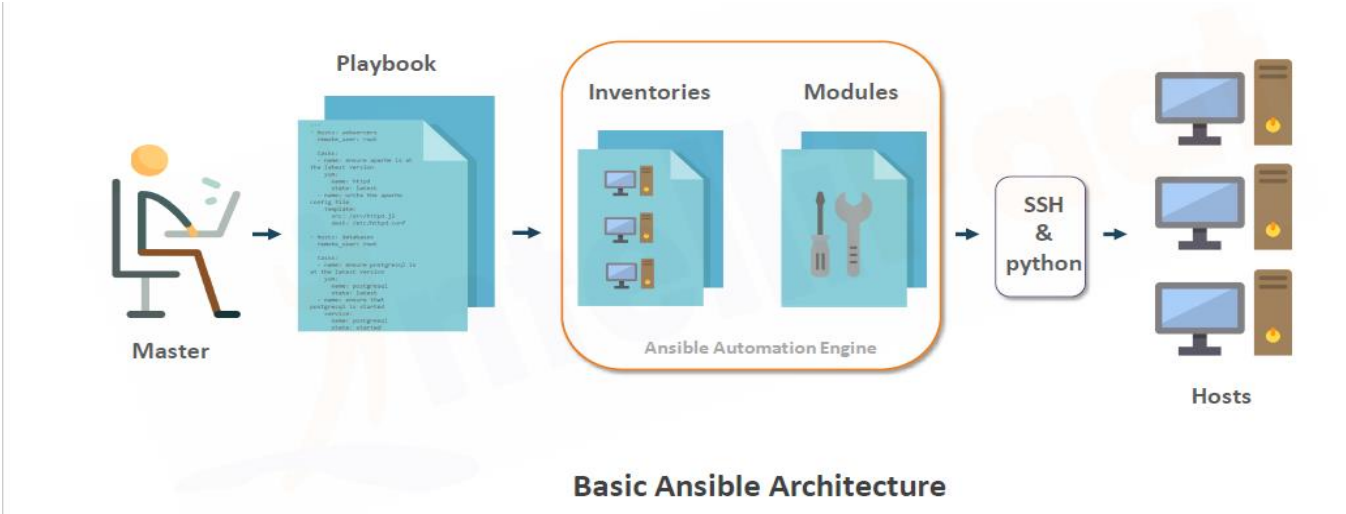

Percussion


Cisco

Twitter

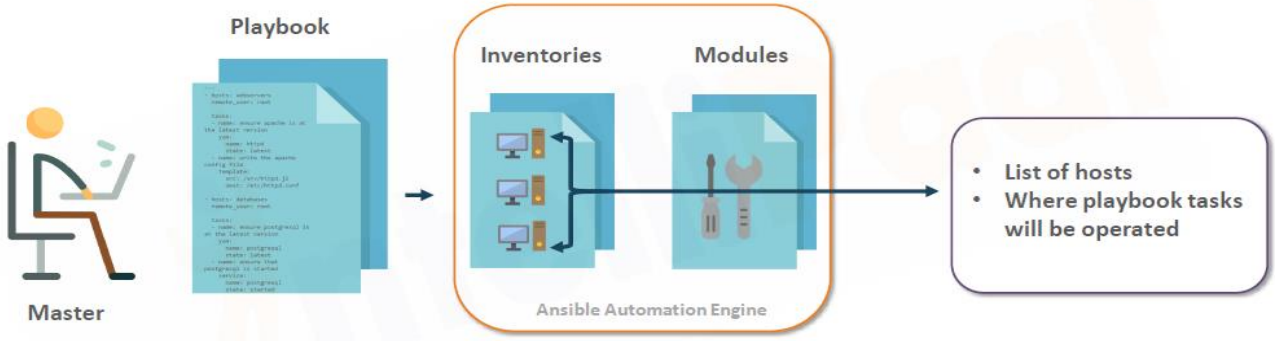
## Ansible Architecture – Master


## Ansible Architecture- Hosts

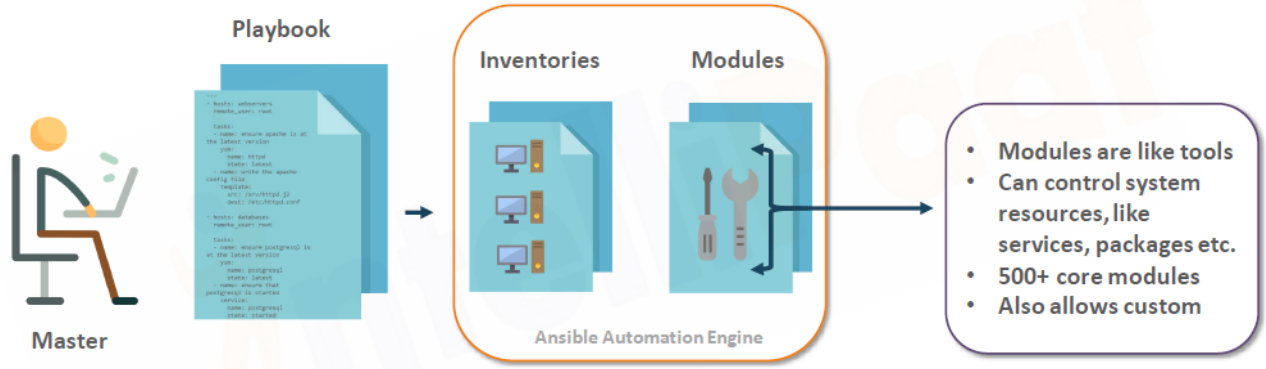



Basic Ansible Architecture

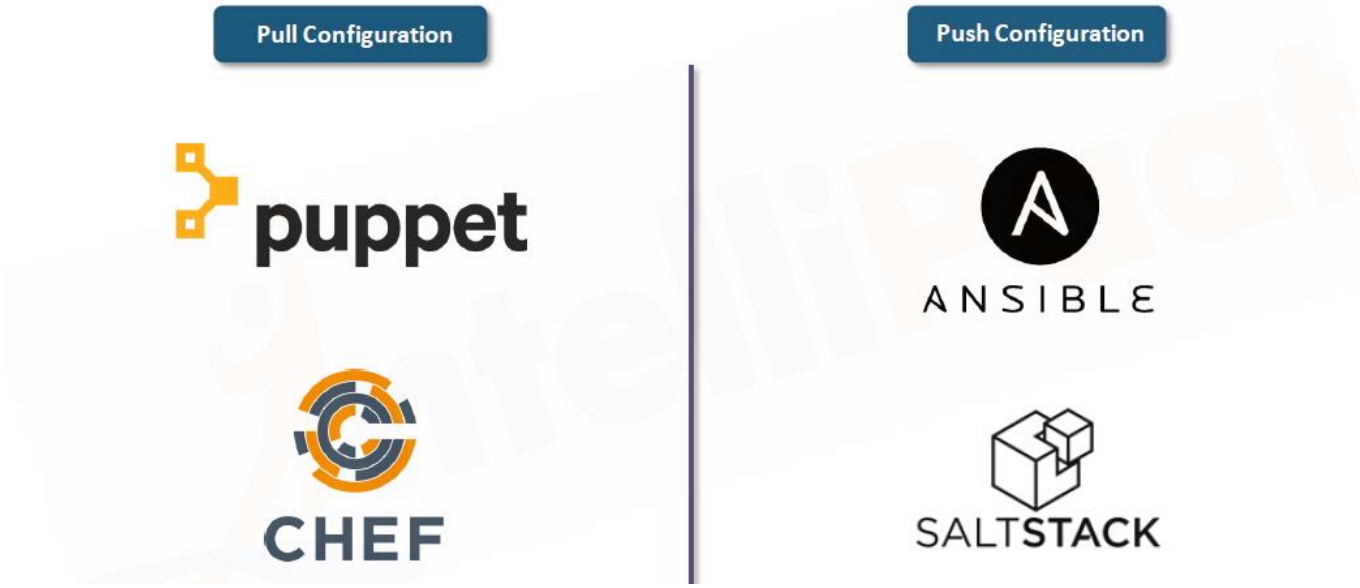
## Ansible Architectue – Inventories



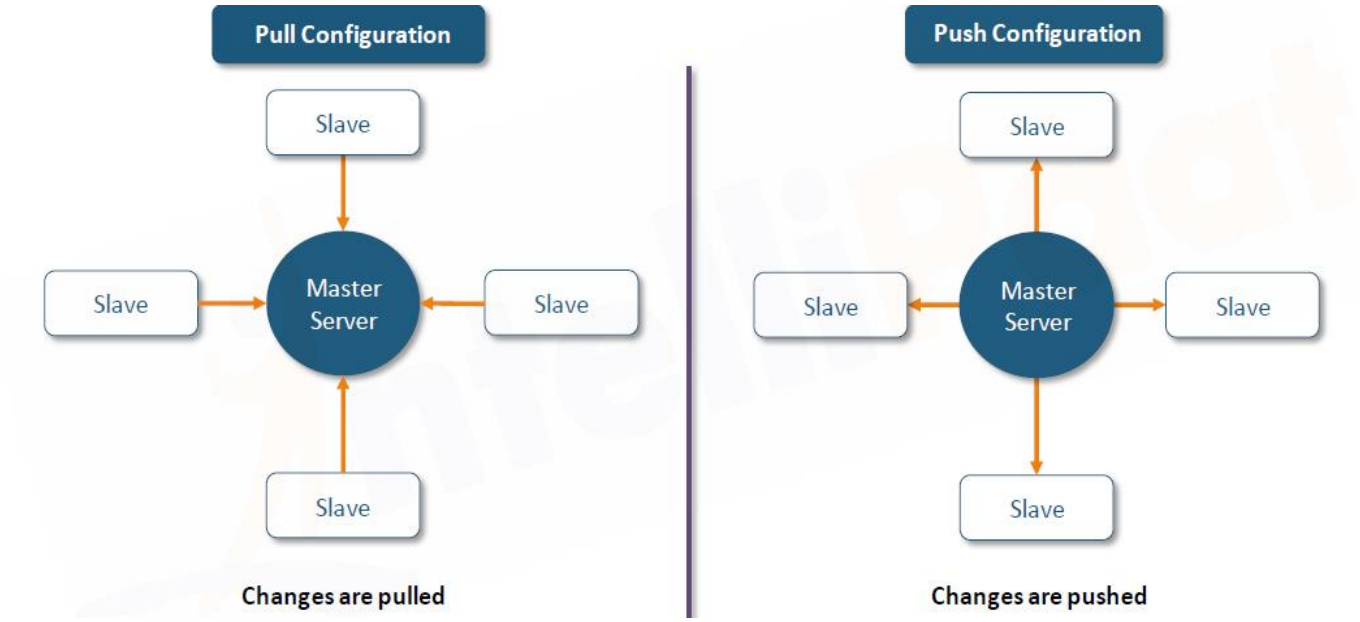

## Ansible Architecture – modules



## Types of Configuration Management Tools




## Between deference push configuration and pull configuration



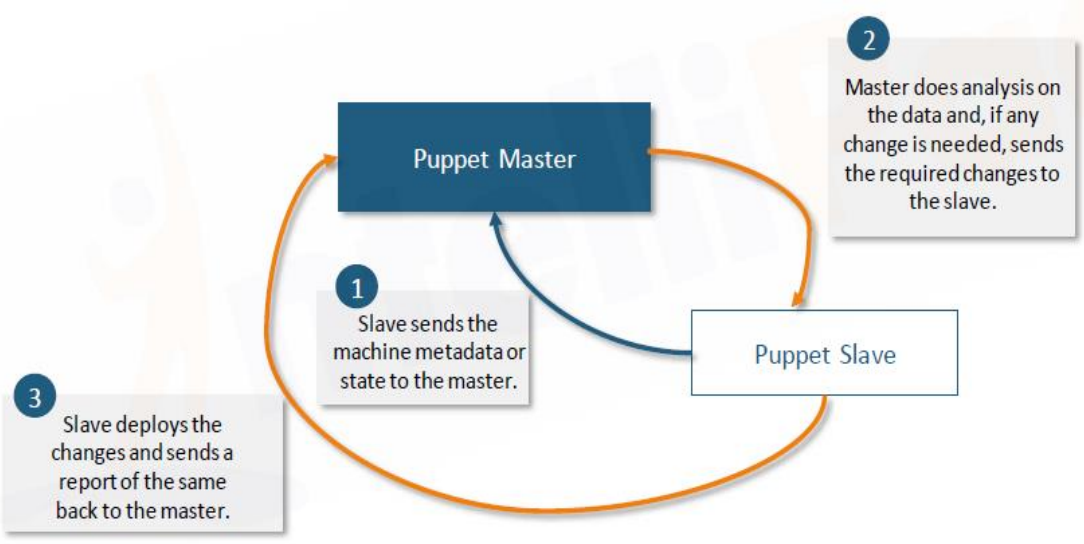
## What is Puppet

**Puppet is an open-source software configuration management tool .**

## Feature of Puppet

- **Large User Base**
- **Big Open-source Community**
- **Documetion**
- **Platform Support**

## Puppet Architecture

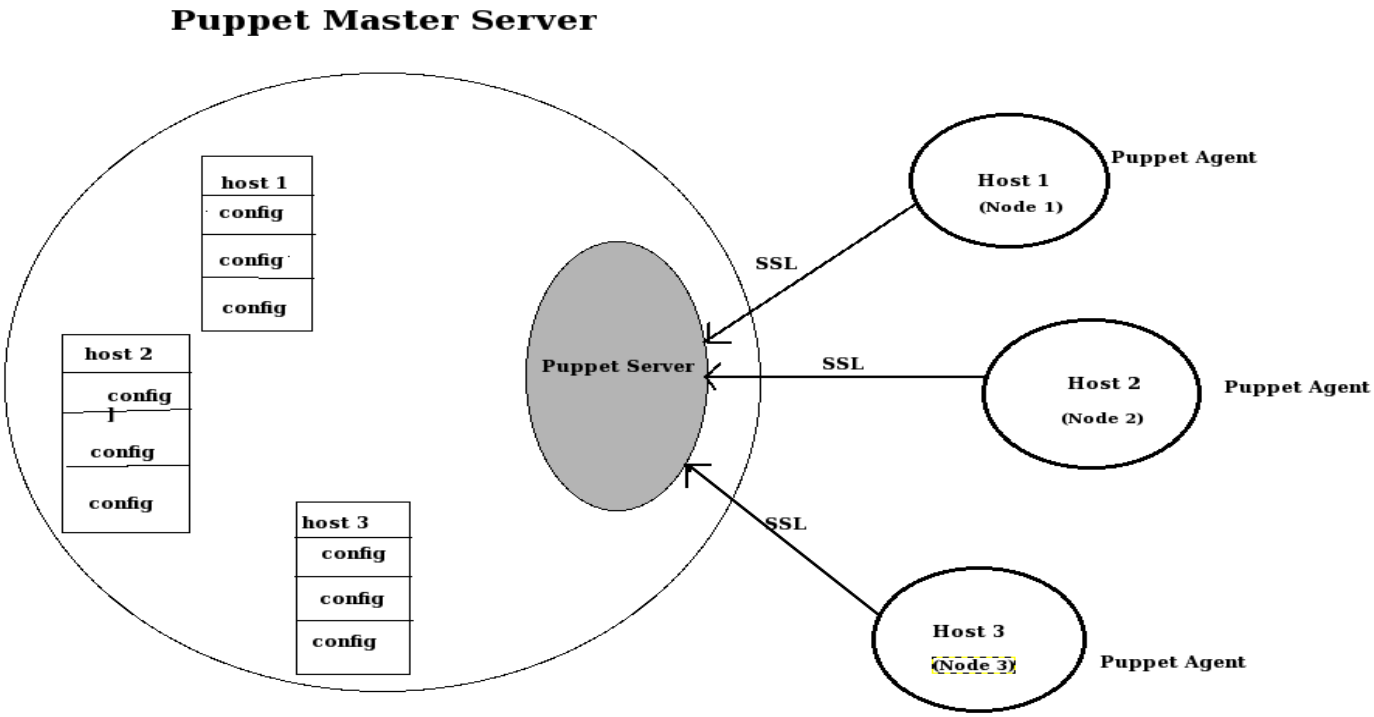



## Puppet Architecture :- SSL Connection (Secure Socket Layer )

**Because Puppet nodes have to ineract with the master , all the information which is communicated between the master node and slaves nodes are encrypted using SSL certificates .**

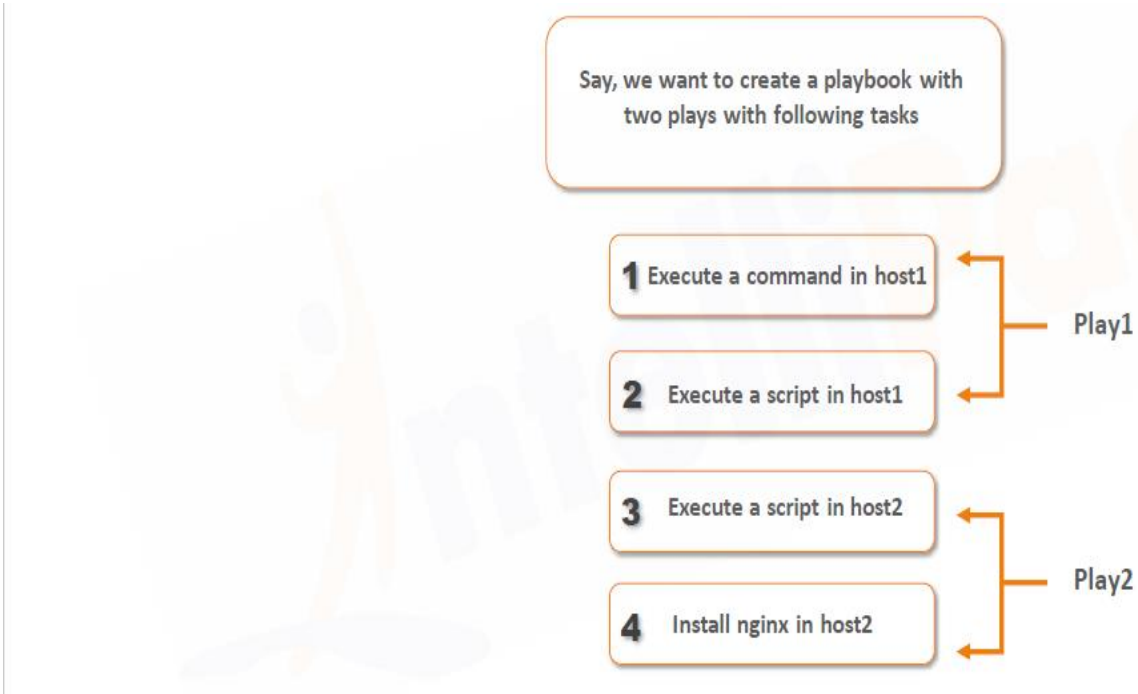
**The certificate signing process is as follows .**


## How does work Puppet



## Ansible Playbook Structure


## Create Ansible Playbook- Example

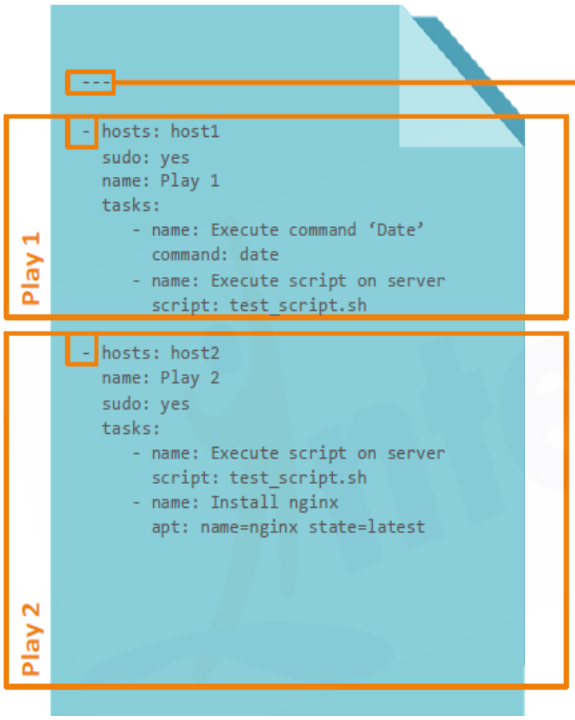


## Creating Ansible Playbook- Example

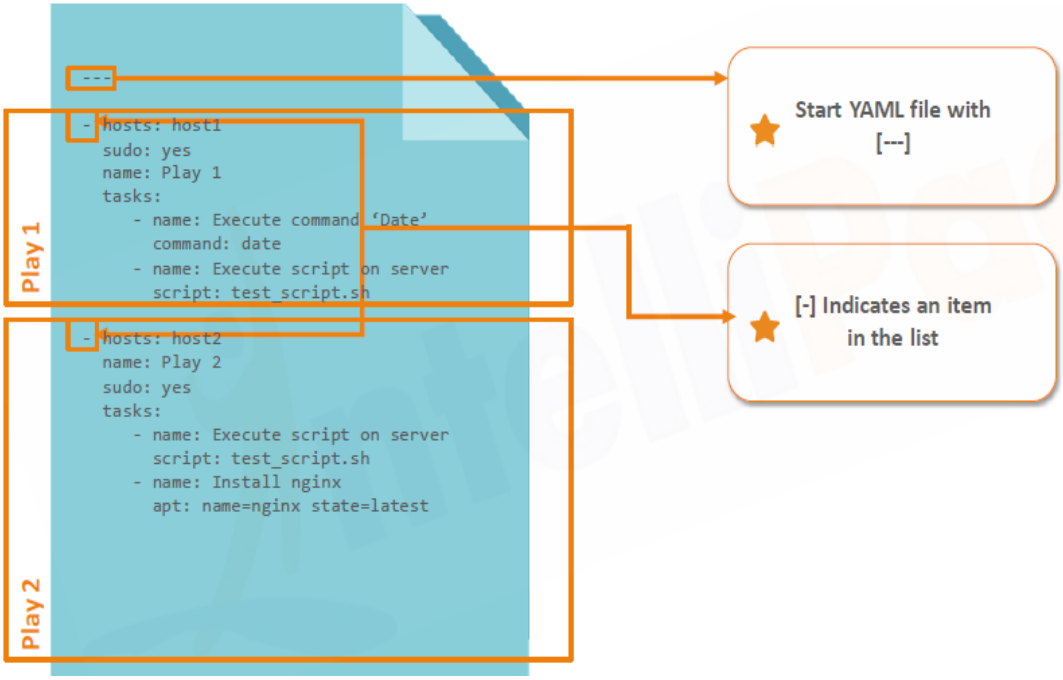
```
---

- hosts: host1
  sudo: yes
  name: Play 1
  tasks:
     - name: Execute command 'Date'
       command: date
     - name: Execute script on server
       script: test_script.sh


- hosts: host2
  name: Play 2
  sudo: yes
  tasks:
     - name: Execute script on server
       script: test_script.sh
     - name: Install nginx
       apt: name=nginx state=latest
```

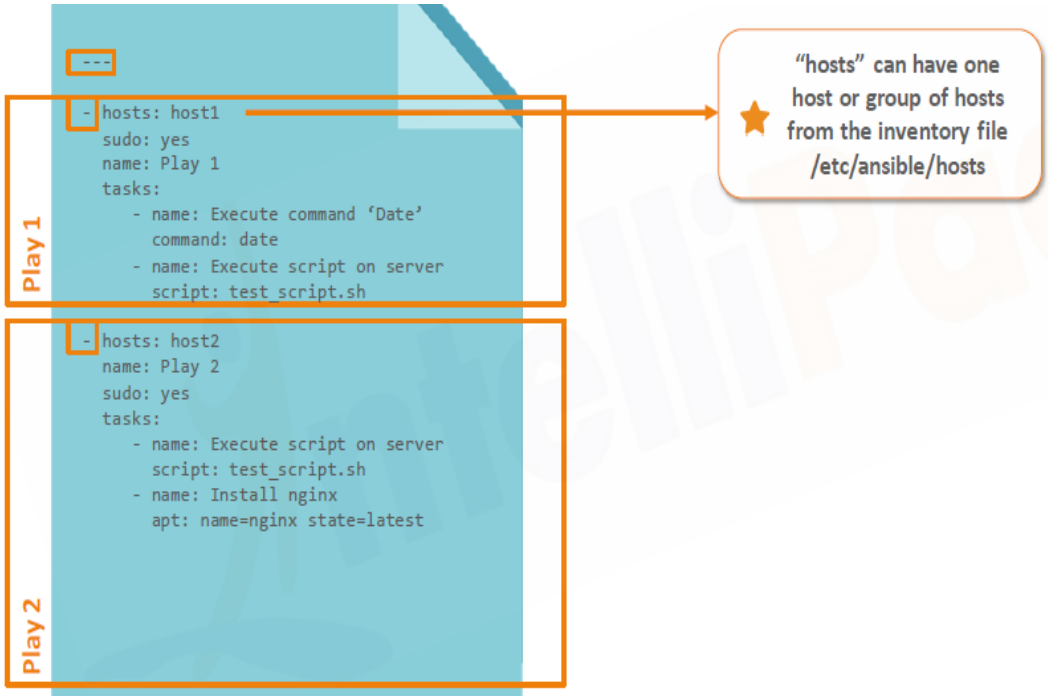
Say we want to create a playbook with two plays with following tasks

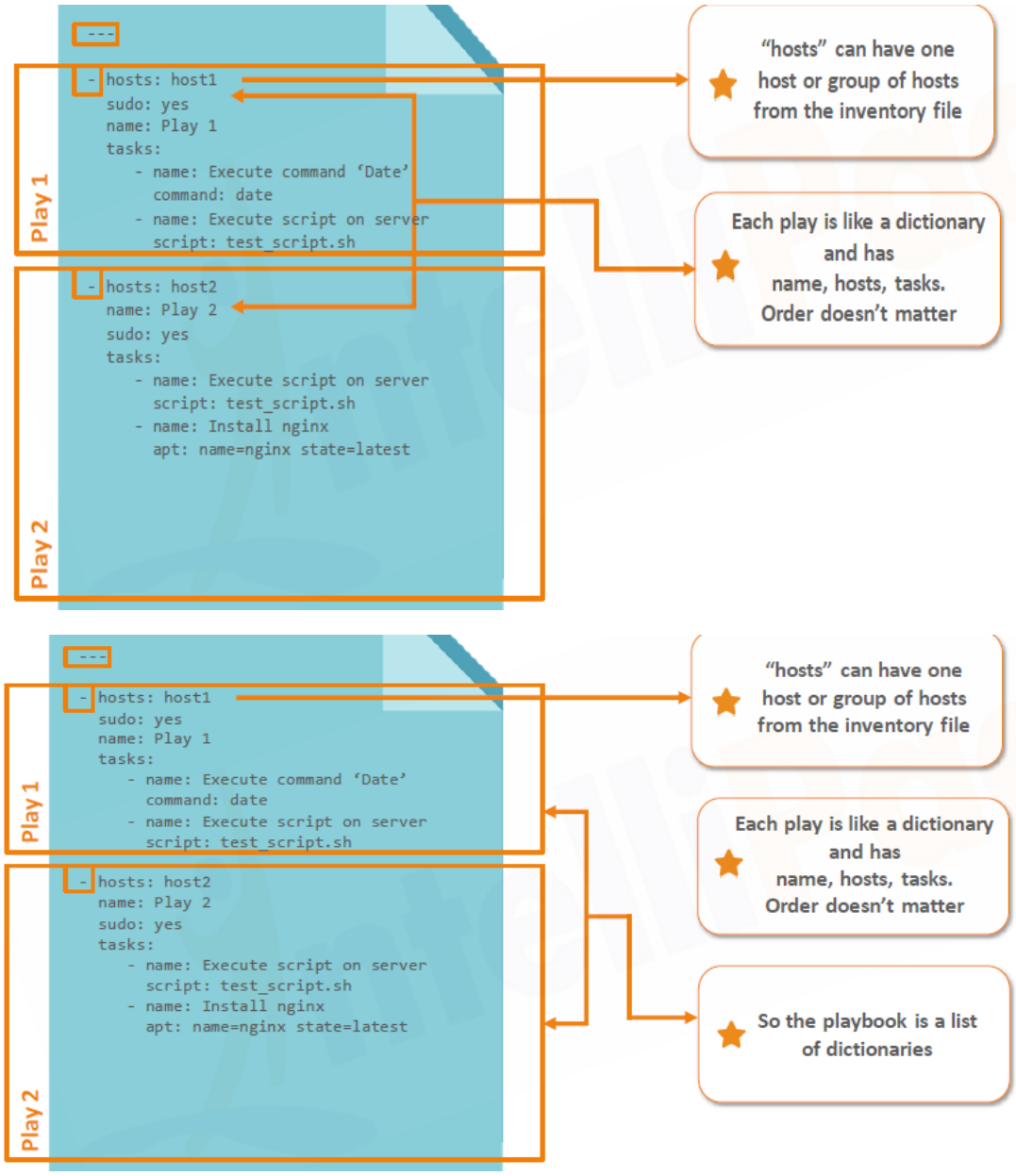
1. Execute a command in host1
2. Execute a script in host1
3. Execute a script in host2
4. Install nginx in host2

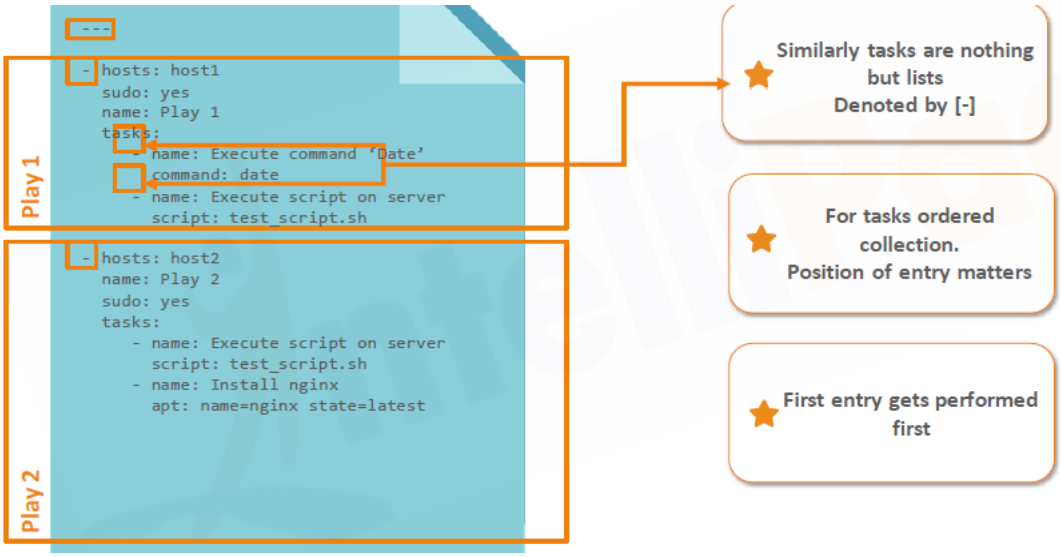



★ Start YAML file with [---]


---
- hosts: host1
sudo: yes
name: Play 1
tasks:
- name: Execute command 'Date'
command: date
- name: Execute script on server
script: test_script.sh
- hosts: host2
name: Play 2
sudo: yes
tasks:
- name: Execute script on server
script: test_script.sh
- name: Install nginx
apt: name=nginx state=latest
Play 1
Play 2
Start YAML file with [---]
[-] Indicates an item in the list


---
- hosts: host1
sudo: yes
name: Play 1
tasks:
- name: Execute command 'Date'
command: date
- name: Execute script on server
script: test_script.sh
- hosts: host2
name: Play 2
sudo: yes
tasks:
- name: Execute script on server
script: test_script.sh
- name: Install nginx
apt: name=nginx state=latest
Play 1
Play 2
"hosts" can have one host or group of hosts from the inventory file /etc/ansible/hosts

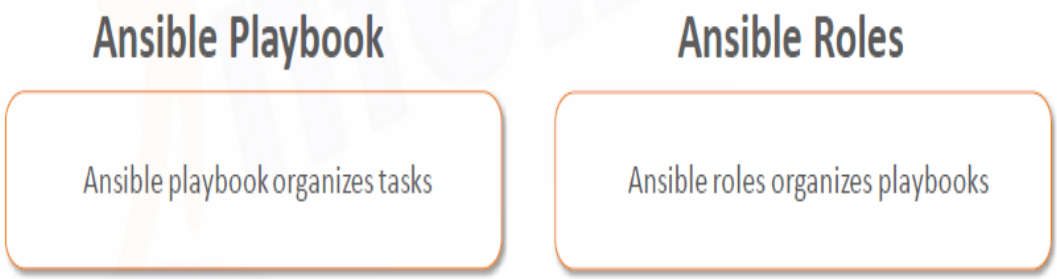

---
- hosts: host1
sudo: yes
name: Play 1
tasks:
- name: Execute command 'Date'
command: date
- name: Execute script on server
script: test_script.sh
Play 1
- hosts: host2
name: Play 2
sudo: yes
tasks:
- name: Execute script on server
script: test_script.sh
- name: Install nginx
apt: name=nginx state=latest
Play 2
"hosts" can have one host or group of hosts from the inventory file
Each play is like a dictionary and has name, hosts, tasks. Order doesn't matter
---
- hosts: host1
sudo: yes
name: Play 1
tasks:
- name: Execute command 'Date'
command: date
- name: Execute script on server
script: test_script.sh
Play 1
- hosts: host2
name: Play 2
sudo: yes
tasks:
- name: Execute script on server
script: test_script.sh
- name: Install nginx
apt: name=nginx state=latest
Play 2
"hosts" can have one host or group of hosts from the inventory file
Each play is like a dictionary and has name, hosts, tasks. Order doesn't matter
So the playbook is a list of dictionaries




## What is Ansible Roles

**An ansible role is group of tasks, files , and handlers stored in a standardlized structure .**

**Roles are small functionalities which can be use indenpendently used but only within playbook .**

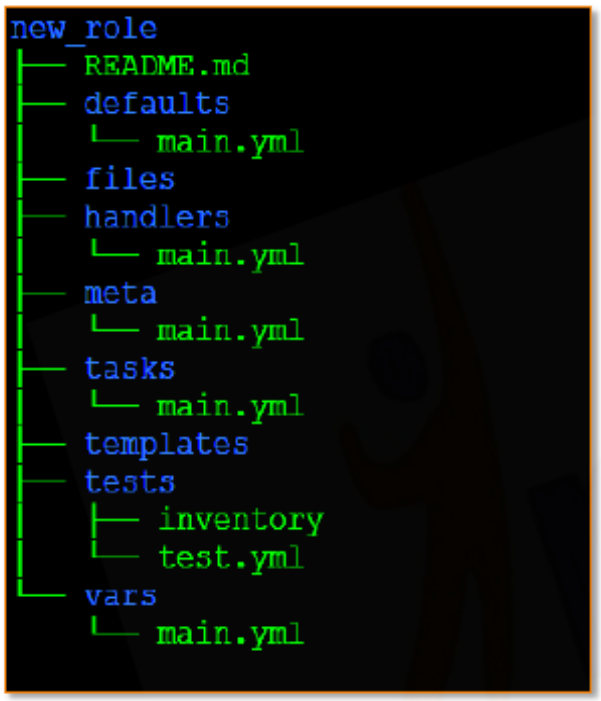



## Why do we need Ansible Roles

- **Roles simplifies writing complex playbooks**
- **Roles allow you to reuse common configuration steps between diifrent types of servers**
- **Roles are flexible and can be easily modified**

## Structure of Ansible Role

**Structure of an ansible role consists of below given components**




Structure of an Ansible Role

**Defaults**: Store data about the role, also store default variables.

**Files**: Store files that needs to be pushed to the remote machine.

**Handlers:** Tasks that get triggered from some actions.

**Meta:** Information about author, supported platforms and dependencies.

**Tasks**: Contains the main list of tasks to be executed by the role.

**Templates**: Contains templates which can be deployed via this role.

**Handlers:** Tasks that get triggered from some actions.

**Vars:** Stores variables with higher priority than default variables. Difficult to override.