Ansible

Ansible is an open-source IT engine that automates application deployment, cloud provisioning, intra service orchestration, and other IT tools.

Ansible is easy to deploy because it does not use any agents or custom security infrastructure on the client-side, and by pushing modules to the clients. These modules are executed locally on the client-side, and the output is pushed back to the Ansible server.

It can easily connect to clients using SSH-Keys, simplifying though the whole process. Client details, such as hostnames or IP addresses and SSH ports, are stored in the files, which are called inventory files. If you created an inventory file and populated it, then Ansible can use it.

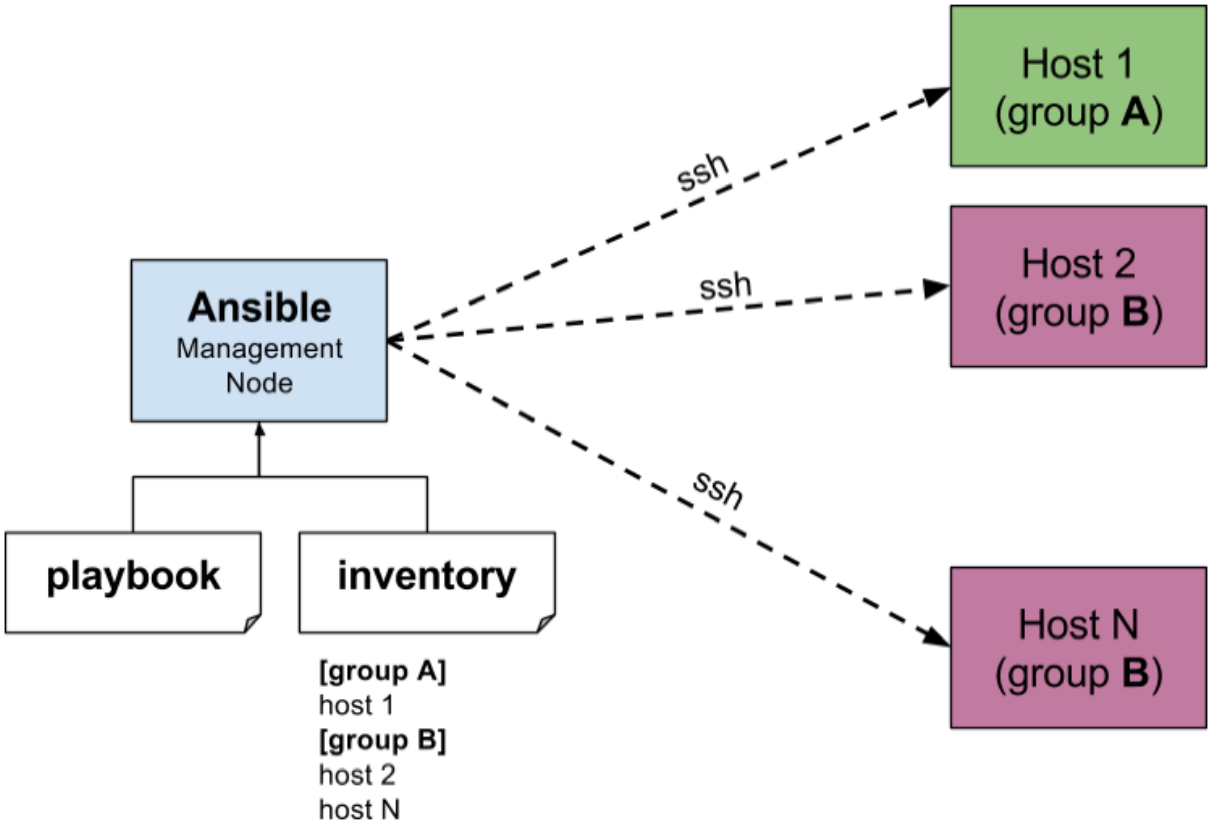
Ansible uses the playbook to describe automation jobs, and playbook, which uses simple language, i.e., YAML. YAML is a human-readable data serialization language & commonly used for configuration files, but it can be used in many applications where data is being stored.

Ansible pushes small programs after connecting to your nodes which are known as "Ansible Modules". Ansible runs that module on your nodes and removes them when finished.

How Ansible Works?

The picture given below shows the working of Ansible.

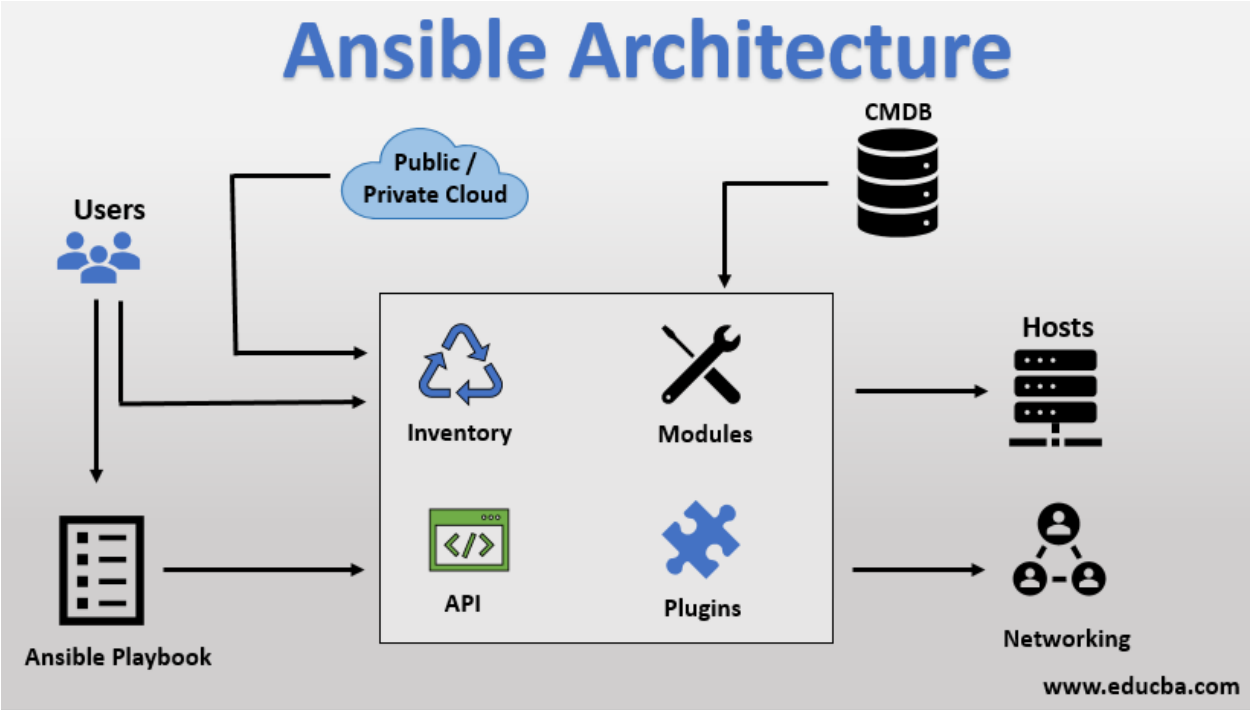
Ansible works by connecting to your nodes and pushing out small programs, called "Ansible modules" to them. Ansible then executes these modules (over SSH by default), and removes them when finished. Your library of modules can reside on any machine, and there are no servers, daemons, or databases required.



- The management node in the above picture is the controlling node (managing node) which controls the entire execution of the playbook.
- It's the node from which you are running the installation.
- The inventory file provides the list of hosts where the Ansible modules needs to be run and the management node does a SSH connection and executes the small modules on the hosts machine and installs the product/software.
- Ansible takes on a modular approach, making it easy to extend to use the functionalities of the main system to deal with specific scenarios.
- Modules can be written in any language and communicate in standard JSON.

Beauty of Ansible is that it removes the modules once those are installed so effectively it connects to host machine , executes the instructions and if it's successfully installed removes the code which was copied on the host machine which was executed.

Ansible Architecture:



Imp:

- Control machine – Machine from where we can manage other machines.
- Remote machine – Machines which are handled/controlled by control machine.
- Modules-The module is a command or set of similar commands which is executed on the client-side.
- Play: Execution of playbook
- Ansible inventories are lists of hosts (nodes) along with their IP addresses, servers, databases etc. which needs to be managed. Ansible then takes action via a transport – SSH for UNIX, Linux or Networking devices and WinRM for Windows system.
- Task===>A block that defines a single procedure to be executed, e.g. Install a package.
- Facts==>Global variables containing information about the system, like network interfaces or operating system.
- Playbook===>The entry point for Ansible provisioning, where the automation is defined through tasks using YAML format.

Installation Steps

1. Make sure your Ubuntu machine is up to date with latest packages.
   $ sudo apt-get update
   $ sudo apt-get install software-properties-common

2. Now we will install the Ansible PPA repository on the system using below command
   $ sudo apt-add-repository ppa:ansible/ansible

3. Install ansible after successfully adding the ansible ppa repository
   $ sudo apt-get install ansible

4. Check for ansible version after installation is done.
   $ ansible –version

5. Generate ssh key in the ansible machine, which we have to copy to all the remote hosts for doing deployments or configurations on them.
   $ ssh-keygen -t rsa -b 4096 -C "srinivas"

6. Copy the ssh key generated to the remote host using the below command
   Note: Before copying the ssh key make sure that you are able to ssh the remote host where you want to copy the key
   $ ssh-copy-id root@192.168.43.181
7. Now we need to edit the "hosts" file of ansible by specifying the group of servers/remote hosts which we need to connect and perform operations on.

   Open the hosts file for editing, I am using nano for editing.
   $ sudo nano /etc/ansible/hosts
   $ sudo nano /etc/hostname

8. Now as we have setup our "hosts" file and other configurations are done, let's try a very simple ansible command, which will ping all the servers listed in the "hosts" file.
   $ ansible all -m ping

Note: check the host list using below command

```
$ ansible  --list-hosts all
```

$ sudo apt-get update

$ sudo apt-get install software-properties-common

```
$ sudo apt-add-repository ppa:ansible/ansible
$ sudo apt-get update
$ sudo apt-get install ansible
```