Introduction about Ansible:

 Ansible is an open source, powerful automation software for configuring, managing and deploying software applications on the nodes without any downtime just by using SSH.

 Most of the IT Automation tools runs as a agent in remote host, but ansible just need a SSH connection and Python (2.4 or later) to be installed on the remote nodes to perform its action.

Here we have install ansible on redhat enterprise linux 7.0 and node the node installation happened at Windows.

Prerequisites:

Python 2.4 since Ansible developed in Python languages.

YML

Steps to install ansible on RHEL and  it’s basic configuration:

**“Yum -y install ansible “**

The above command will install bunch of python dependencies, after installation gtes complete, check ansible version by using  command.

**“Ansible --version”**

It will show version of ansible installed on.

We have created basic configuration now, ansible directory structure.

/etc/ansible by default, which store main configuration all ansible config.

“/etc/ansible/hosts”  this file store information about hosts and hosts groups.

“/etc/ansible/ansible.cfg” the main config file for ansible.

“/etc/ansible/roles” This folder allows you to create folders for each server role, web/app/db, etc.

 Ansible need to know your remote server name /ip address to configure ,this info will stored on hosts file.

How ansible works:

Ansible categorize into two types of server:

1. controlling machine

     2. nodes

The controlling machine, where Ansible is installed and Nodes are managed by this controlling machine over SSH.

The location of nodes are specified by controlling machine through its inventory.

The controlling machine (Ansible) deploys modules to nodes using SSH protocol and these modules are stored temporarily on remote nodes and communicate with the Ansible machine through a JSON connection over the standard output.

**Ansible is agent-less**, that means no need of any agent installation on remote nodes, so it means there are no any background daemons or programs are executing for Ansible, when it’s not managing any nodes.

Ansible can handle 100’s of nodes from a single system over SSH connection and the entire operation can be handled and executed by one single command ‘ansible’. But, in some cases, where you required to execute multiple commands for a deployment, here we can build playbooks.

Playbooks are bunch of commands which can perform multiple tasks and each playbooks are in YAML file format.

SSH agent configuration from controlling machine to hosts:

  If yours hosts are running on Linux , we will generate SSH key for hosts and copy it on hosts server’s.

Steps to generate SSH key:

$ ssh-keygen -t rsa -b by using this command will generate ssh key , it will ask for password provide password here.

One ssh key generate successfully, now we have to copy generated key on our hosts VM’s

$ssh\_copy-id server@host\_name

After copying all SSH Keys to remote host, now perform a ssh key authentication on all remote hosts to check whether authentication working or no?

**Inventory file:**

Inventory file hold hosts info like which host we need to connect from local to remote host, by default inventory file lies under /etc/ansible/hosts

Open hosts file in vi/vim editor

Vim /etc/ansible/hosts

Provide hosts detail under hosts like below.

[web server]

10.102.84.11

10.102.84.164

Save it.

Check if u r accessing hosts vm’s with command

“**Ansible -m ping [web server]”**

we are using another module called ‘command‘, which is used to execute list of commands (like, df, free, uptim, etc.) on all selected remote hosts at one go, for example watch out few examples shown below

To check partitions of all remote hosts.

Eg: **ansible -m command -a “df-h”  web server**

To check memory usage and uptime of all server

Eg: **ansible -m command -a “free-mt” web server**

**Ansible -m command -a “uptime” web server.**