ANSIBLE DOCS

Ansible Overview -- IT automation, config management tool.It uses playbooks to deploy, manage, build and test the servers.

Configuration Management Tools

Idempotence ---- If repeated applications has the same affect as a single apps.

It will always maintain the state…

What is Ansible? ----- Managing the nodes through One server to control it is called Ansible.

Server/Master/Control-node ------ Nodes/Slaves/Manage-Node

Two machines who can start the communications first – In Ansible Master starts the communication i.e Push mode or mechanisms…

Ansible is agentless – which is no need software needed to install in nodes, but it need python for communication.

Need YAML --- yaml will convert to python and execute in nodes by ansible…

Ansible --- REDHAT

Ansible --- change management, provisioning(Infrastructure), Automation, Orchestration…

No agents in production , No database, No residual software, No complex upgrades…

Ansible is a lightweight software….

Ansible its like software once start it will run, but chef is like service its running in background…

Communication --- ssh , winrm..

Built in security --- users ssh, Sudo privilege, encrypted vault, no pki needed…

Competitors – chef, salt,fabric,puppet…

###Installing Ansible in Control Server####

Install in Ubuntu

$ sudo apt update

$ sudo apt install software-properties-common

$ sudo add-apt-repository --yes --update ppa:ansible/ansible

$ sudo apt install ansible

After the commands exe the Ubuntu need user to create then access the path with sudo privilege then the default ansible.cfg file for 2.12 will not contain default content for the refer the content in 2.12 then copy the 2.9 example content..

###For Managing nodes ###

Use the shell script which will create user and passwd then allow sudo privilege then allow passwd auth.Wll refer the own github for shell script..

####For Ubuntu########

#!/bin/bash

####Create user and set passwd in ubuntu#######

username=ansible

passwd=Test@1234567

useradd $username

echo -e "$passwd\n$passwd" | passwd "$username"

####Allow sudo privilege########

sed -i -e '/^root/ a ansible ALL=(ALL:ALL) NOPASSWD: ALL' /etc/sudoers

####Enable Password authentication for ansible user###

sed -i -e '/^PasswordAuthentication/ s/no/yes/' /etc/ssh/sshd\_config

####Restart the service of ssh#######

sudo service sshd restart

After that edit the config file /etc/hosts

And add the private ip or hosts of the manage nodes..

####Core components of Ansible

1. Inventories -- list of all nodes. Default and own created files. Can be static and dynamic
2. Modules – It is a tool. Module can done the activities what ever in Playbooks or adhoc. Modules are idempotent.
3. Variables --- To use dynamically or repeateadly we can use variables..it refers jinja.
4. Facts --- it is a way of getting data from systems. Use this facts in playbook variables.Speed up the execution….
5. Playbooks and play ---- A play is a task.. playbooks are in yaml formats.. playbook is made up to individual plays…
6. Configuration files --- we can disable or enable options in config file… use the config files other than default eg.. ANSIBLE\_CONFIG(an environment variable, ansible.cfg (current dir), .ansible.cfg (in home dir)…Config file is read when the playbook is run…
7. Templates --- There is an ansible modules called templates..
8. Roles and Handlers -- handlers whenever restart the service using handlers. Reusablity is roles..
9. Ansible Vault -- it is a secure store. Encrypted files.
10. Ansible Tower – An enterprise way of using GUI…

In inventory file swe can grouping with tags

Ex – [webserver]

Ip1

Ip2

For default inventory calling using # ansible all –m ping

For own inventory - #ansible all –i <inventory file> -m ping

# man ansible to check all ansible commands…

#######YAML #####

Its not a code , it is a human and machine readable…

Yet another markup language

Yaml aint markup language

Its starting with 3 ---

If starts with one – it is a list item..

####List of details starts with single –

---

- Devops

- AWS

- Azure

For example if I want to show one employee details

---

Name: sarvan

Age: 28

Address: Royapuram

Online ---- yaml validators

---

- name: sarvan

address: royapuram

- name: sarvan2

address: royapuram2

---

- name: sarvan

address: royapuram

vehicles:

- cycle

- bike

courses:

- Name: Devops

Date: 12-12-12

- name: sarvan2

address: royapuram2

vehicles:

- bike

courses:

- Name: Devops

Date: 11-11-13

- Name: AWS

Date: 23-12-12

In YAML space maintainance is very important….

Modules are very important to write playbooks..

####Install apache in Ubuntu using playbooks##

Apt update

Apt install apache2 –y is the command

##To create playbook for the cmds

---

- hosts: ubuntu

  become: yes

  tasks:

  - name: update the machine

    apt:

      name: update

      update\_cache: yes

  - name: This will install apache2 in ubuntu

    apt:

      name: apache2

      state: present

For asking passwd while running playbooks –k is the key for ask passwd…

-K caps k is the –ask-become sudo privilege (--become)

# ansible-playbook ubuntuonapache.yml –K --it will ask for passwd no need to enter in playbook as a root.

###Install apache inRedhat###

#yum install httpd

---

- hosts: redhat

  become: yes

  tasks:

  - name: Install apache server in redhat

    yum:

      name: httpd

      state: present

  - name: Start the service of httpd

    service:

      name: httpd

      state: started

Some of the popular modules

1. User accounts
2. Groups
3. Command
4. Copy module
5. Get url
6. Htpasswd
7. Lineinfile
8. Ping
9. Script module
10. Service module
11. Shell module
12. Unarchive module
13. Yum

###Install tomcat in Ubuntu using simple commands##

#apt install tomcat9

---

- hosts: ubuntu

  tasks:

  - name: Tomcat 9 is installing in ubuntu20

    apt:

      name: tomcat9

      state: present

      update\_cache: yes

#ansible-playbook tomcat.yml –K

######USER MODULE######--only linux user###

Create user and group using modules

---

- hosts: all

  become: yes

  tasks:

    - name: create userin all machines

      user:

        name: tomcat

        state: present

    - name: Create one group

      group:

        name: tomcat

        state: present

####COMMAND Modules###########

Command doesnot guarantee for idempotent

Command to execute the raw linux statement and the idempotent is not working as per the commands

#maximum try to avoid this modules###

Commands have arguments so use –a in ansible adhoc commands

#ansible all -vv -m command -a "cat /etc/passwd" --(-vv will show what will happen in the command during execution…

(there are –vvv, -vv, -v)

#ansible ubuntu -m setup ---- will show all the details of the systems this is a facts…

One option called

Gather\_facts: no -- will not show the facts while executing…

###########COPY MODULE################

Using the copy module and command module create one script and copied to manage node and execute it using command module

---

- hosts: redhat

  become: yes

  tasks:

    - name: Copy Module

      copy:

        src: /home/ansible/playbooks/rhuser.sh

        dest: /home/ec2-user/rhuser.sh

        mode: '0777'

    - name: Execute the shell script

      command: sh /home/ec2-user/rhuser.sh

      become: yes

###Get\_url modules########

Getting url for download Jenkins war file from url to remote systems

---

- hosts: redhat

  become: yes

  tasks:

    - name: Download the content from url to remote system

      get\_url:

        url: https://sg.mirror.servanamanaged.com/jenkins/war-stable/2.375.2/jenkins.war

        dest: /home/ec2-user/jenkins.war

#####Setup modules and then filter it using adhco and playbook######

# ansible -m setup -a "filter=ansible\_os\*" Ubuntu

####This will filter the particular thing in setup####

####CONDITIONALS in Playbooks#############

In this scenario I want execute both yum and apt module in same playbooks for that im using conditional which is called “WHEN”

---

- hosts: all

  become: yes

  tasks:

    - name: install apache server in redhat

      yum:

        name: httpd

        state: present

        when: ansible\_os\_family == "RedHat"

    - name: install apache in ubuntu

      apt:

        name: nginx

        state: present

        when: ansible\_os\_family == "Debian"

###########HANDLERS############################

In tomcat server deploy war file day by day and need to restart everytime after deploy for that using handlers…

---

- hosts: ubuntu

  become: yes

  tasks:

    - name: install tomcat 9

      apt:

        name: tomcat9

        state: present

      notify:

        - restart tomcat

    - name: copy some files in remote

      copy:

        src: /home/ansible/playbooks/test

        dest: /tmp/test

      notify:

        - restart tomcat

  handlers:

    - name: restart tomcat

      service:

        name: tomcat9

        state: restarted

Task which notifying called handlers ---

Listening is introduced newly and notification is old in ansible which start earlier..

Listen can be used in when we need to restart multiple service using handlers,then use listening….

###Variables in ansible##########

Variables can be defined in two files one is inventory file and playbooks.

In inventory there are group var and host vars.

##Create group\_vars (this will be located in inventory files location)

Cd group\_vars and create one file Ubuntu…

---

- hosts: ubuntu

  become: yes

  tasks:

    - name: install tomcat 9 in ubuntu machines

      apt:

        name: "{{ packagename }}"

        state: present

      notify:

        - restart tomcat

    - name: copy files to remote

      copy:

        src: /home/ansible/playbooks/new

        dest: "{{  destcopy  }}"

      notify:

        - restart tomcat

  handlers:

      - name: restart tomcat

        service:

          name: "{{ packagename }}"

          state: restarted

####Ansible Conditional####

With items

Install 3 or more packages using with items

Loop is with items

---

- hosts: amazon-linux

  become: yes

  tasks:

    - name: Install utilities

      yum:

        name: "{{ item  }}"

        state: present

      with\_items: ["git,nano,tree,wget"]

#####LINEINFILE############

---

- hosts: amazon-linux

  become: yes

  tasks:

    - name: first copy the content from server to remote

      copy:

        src: /etc/ansible/ansible.cfg

        dest: "{{ destcopy }}"

    - name: Edit the file using lineinfile

      lineinfile:

        path: /tmp/ansible.cfg

        regexp: '^[colors]'

        insertafter: '^[colors]'

        line: '#New color is RED'

#######TEMPLATES IN ANSIBLE########################################

Jinja2 templates –this is carried out in python.

Templates also using source and destination but as a dynamic files.

Playbook is same for static jinja2 templates,

Create and basic jinja2 templates with some content

#vim test.j2 ---content hello hi

And then mention in playbooks..

---

- hosts: all

  become: yes

  tasks:

    - name: using basic templates to copy static content

      templates:

        src: test.j2

        dest: /tmp/test

##For dynamic content we need to know some jinja syntactical templates…

For adding {{}} this in jinja file and mention hostname which will be dynamically changing as per the hosts in the below code…For that first create the jinja file ..

---

- hosts: all

  become: yes

  tasks:

    - name: using dynamic templates

      template:

        src: firsttemplate.j2

        dest: /tmp/firsttemplate

Create variables in group and host dir then create the jinja2 template with calling the name

---

- hosts: all

  become: yes

  tasks:

    - name: using dynamic templates

      template:

        src: firsttemplate.j2

        dest: /tmp/firsttemplate

    - name: second template dynamic

      template:

        src: second.j2

        dest: /tmp/second

###Filters is very important in jinja templates follow some useful filters in the docs

<https://docs.ansible.com/ansible/2.8/user_guide/playbooks_filters.html#id8>

When if no modules is available for particular activity use command.

#DEBUG#####

Command module again checking

While executing command modules I want show the output while running the playbooks for that im using ‘register’

Whenever creates register it will be variables no need to create variables individually..

Debug and register is used for commands modules while showing output and error what ever if the command….

---

- hosts: amazon-linux

  tasks:

    - name: execute the linux commands

      command: hostname -f

      register: cmd\_content

    - name: print variables contents

      debug:

        msg: "{{ cmd\_content }}"

register ---“cmd\_content”

if register is there then msg will available in debug…

Predefined variables are given by ansible is FACTS

Whenever if we write anything in when it will redefined it..no need to use {{}}.

---

- hosts: amazon-linux

  tasks:

    - name: execute the linux commands

      command: hostname -f

      register: cmd\_content

    - name: debug statment for OS

      debug:

        msg: "OS family is  {{ ansible\_os\_family }} so will skip the operating system specified"

    - name: print variables contents

      debug:

        msg: "{{ cmd\_content.stdout }}"

      when: cmd\_content.stderr  == ""

    - name: print command error if present

      debug:

        msg: "{{ cmd\_content.stderr }}"

      when: cmd\_content.stderr != ""

    - name: inventory hostname

      debug:

        msg: "System {{ inventory\_hostname }} has uuid {{ ansible\_product\_uuid }}"

##Check is the command is a dry run which can check the playbook

#ansible-playbook –K playbook.yml –check

And also we can apply which content in playbook need to check is

check\_mode**:** yes it will not execute the content…

is the option in playbook can execute it…

###Interactivity means we can set the passwd authentication like##

**vars\_prompt:**

**-** **name:** my\_password2

**prompt:** Enter password2

**private:** true

This will ask passwd while execute the playbooks….

#######REUSABILITY#######

Reuse the playbooks wherever

Want to run 2 playbooks in one playbook…

###IMPORT AND INCLUDE THE TASK####

Include is a a first way that call one plybook in anther playbook

Import – it executes all the things need to validate

If we have playbookA and playbookB import will check both while exe and if any error in playbookB then through error but include will not validate the playbooks…

####ROLES###############

Files, templates, tasks, handlers, vars, defaults, meta….

To create all dir in roles use below

#ansible-galaxy init hello

Create folder structure properly

Lamp

Site.yml

├── group\_vars

├── host\_var

├── hosts

├── roles

│ ├── Readme.md

###site.yml is a main program which we called other programs

---

- hosts: all

become: yes

roles:

- common

- webserver

###we can use already used roles ####

Galaxy.ansible.com ---url

Here we can find lots of roles package and find it to add in our main playbook simply mentioned it and execute it or else go to the github repo of the code and we can copy the code usinh=g the command

#ansible-galaxy collection install sky\_joker.zabbix

It will download the roles to the server and also path is

~/.ansible/roles/

###In meta folder check the yaml file which has the dependencies to download the roles..

ansible site.yml –url

##In case if the roles is not available in ansible galaxy we can get it from github and mentioned in playbook

####tags

Tags can be helpful for each content in playbook we can skip the first tags and run the second content then we can use the tags….

##using –skip tags###

This will help us to run particular portion in the playbook…