Ansible modules: -

Ansible is an idempotence it means irrespective of num of executions it will give same result. For example, if we run the playbook for no of times and if there is any repetitive tasks it will give same result

Sudo yum install ansible -y

ansible -i inventory ungrouped --list-hosts (it will list hosts which are not in group)

ansible -i inventory web --list-hosts (it will list hosts which are in web)

We no need to install anything in the node we just need to install ansible in the server and from server we can execute below commands

ansible -i 3.90.62.61, all -e ansible\_user=centos -e ansible\_password=DevOps321 –m ping (to connect to node and ping from server)

ansible -i 3.90.62.61, all -e ansible\_user=centos -e ansible\_password=DevOps321 --become -m yum -a "name=nginx state=present" (to install nginx)

ansible –i 3.90.62.61, all –e ansible\_user=centos –e ansible\_password=DevOps321 –-become –m service –a “name=nginx state=started”

become = it will become root user

state = started means it will start the installed nginx package

ansible –i inventory cart - -list-hosts (it will list hosts related to cart in inventory)

ansible –i inventory web - - list-hosts (It will list hosts related to web in inventory)

ansible –I inventory roboshop - - list-hosts (it will list list of hosts mentioned under roboshoup ex: web cart)

ansible –i inventory ungrouped - -list-hosts (it will list host which are not grouped)

ansible –i inventory all - - list-hosts

running ansible play books in server by mentioning hosts in inventory:

ansible-playbook –i inventory –e ansible\_user=centos –e ansible\_password=DevOps321 01-Playbook.yaml

ansible-playbook –I inventory –e ansible\_user=centos –e ansible\_password=DevOps321

02.nginx.yaml

Task level variables can override play level variables, id same variable is available in play and task it will select task level variable only.

This is like in heritance – if we got some property from our four fathers we can keep same property or else we can do our wish

If we want, wen can keep variables in a separate file and we can call that file in our code

We will save variable file as variables.yaml

We will call it in a code by mentioning

vars\_files:

* Variables.yaml ( Ref: 06-variables-from-file)

Ways of providing variables in ansible

Keeping variables in the same code file

Keeping variables in different file and calling that file to the code

Keeping variables in inventory and calling to the code

Providing variables in the run time

Variables preferences

1 CMD

2 task level variables

3 file level variables

4 promt variables

5 play level variables

6 roles

Data types in ansible

We have Key value pair – it is like

Name: srikanth

We have list data types

Skills:

* Skill1
* Skill2
* Skill3

We have map data types

Experience

DevOps: 3

Aws: 2

IT: 2

We have Boolean

Trainer: true

Conditions in Ansible

We have when condition in ansible we can use it when it is exit ok not exit create

For example checking roboshop user available or not

When user.rc !=0 (This condition run below the user creation module)

In above rc is used to check status code and exit status

Loops in Ansible

We have loops in ansible for example adding or deleting any packages

We will write a loop as below and we pass this loop as an {{item}} in the task level

Loop

* Git
* Mysql
* Postfix

Another method of loops

* {name: ‘git’, state: ‘present’}
* {name: ‘mysql’, state: ‘present’}
* {name: ‘postfix’, state: ‘present’}
* {name: ‘nginx’, state: ‘present}

For any module in ansible (add, copy, package , service) we can check in the ansible official documentation by module name and if we don’t have module by ansible we can use builtin.shell or command module and pass value directly next to command module

Ansible machine is called as ansible controller or server

In ansible based on modele state (present , update , start) will be changed

Shell vs command:

Shell is like it will go inside the server and work or execute and shell will work if there is redirections

Command module is like it is working out side of the server and so if there is any redirections it will not work

dnf is for centos

package module is for all

check all modules available in ansible and get knowledge on all modules

we has maximum ansible official modules if not we can use community modules

if we give & >> /dev/null the ansible job will run in background

if we give nohup logs will be running in nohup directory

nobup ansible-playbook -vvv –i inventory.ini –e ansible\_user=centos –e ansible\_password-DevOps321 mysql.yaml & >> /dev/null

-vvv we will get the full info about the logs

Idempotence – if any already created and available it will skip and wont fail

Ansible Roles:

It is a DRY process don’t repeat yourself, it is a perfect folder structure we need to provide folder structure correctly.

We call variabls in ansible as {{component}} we need to create like below folder

Roboshop-ansible-roles

Roles- cart is a folder in cart we have sub folders files tasks and templates

In file we will have cart. Service (we will have service files)

In Task we will have main.yaml for cart

We can create common folder and keep if there are any common tasks in the common folder files and tasks folders and that will be called to main main.yaml based on component we provide

Main.yaml

inventory.ini

while executing any module we will refer which component to run so that the particular component will execute

Example: -

ansible-playbook –i inventory.ini –e ansible\_user=centos –e ansible\_password=DevOps321 component=cart main.yaml

the above command will execute main.yaml from roboshop-ansible-roles and as we passed component as cart it will go to roles folder under roboshop-ansible-roles and check cart folder and execute the cart as we passed component as cart and if there is any modules passed in common it will go and check in common folder also and if and service file is there it will take that file from file folder under cart

ansible templates:

In ansible we can use templates also like variables if any reusable we can add in templates and keep if the values are useable for all roles we will keep In available for all folders and if only useable for particular folder we will keep in that folder itself, usually for template files we give .j2 as an extension

Config in ansible:

If we need to keep inventory.ini or user name we can keep in config file but can keep password in config file as it is not supported

We can keep password in inventory.ini as an [all:vars] by mentioning ansible\_password=DevOps321

If we have different users and passwords we can save them differently

After keeping inventory.ini and username and password we can run as below command

Andible-playbook –e component=cart main.yaml

Handlers in ansible:

In ansible we can use handlers in restarting any service, example if we have and configuration change in server and if any restart required for nginx we need to do manually or if we mention in script every time we run it will restart

But if we enable handler and keep handler file It will automatically detect changes and restart

Ansible tags:

We can use tags In ansible, for example if we need to run particular task then we can create tags and tags name to it. When we run ansible command we need to pass that tag name so that that tag name available tasks only will run

Ansible-playbook –i tags DevOps component=cart main.yaml

Ansible valut:

ansible valut is used to encrypt our target server user credentials

we need to create an vault to the play book and we need to provide a valut password

after that we need to edit that valut with vim and provide our credentials like

aws\_user=centos aus\_password=DevOps321

Ansible valut is not that accurate will need to create individual valut for individual play book which is difficult so we are migrating to aws ssm parameter or secret manager to store out keys

Fork in ansible:

If we are deploying configuration to many number of servers and if there is any chances of server break we can provide –f command to ansible command module and we can specify –f 5 (this –f 5 will run 5 -5 servers at a time and deploy the configuration)

Ansible is used to deploy configuration and also used to deploy application