[day-16] Ansible - playbooks, playbook (1)

→ hosts: all
max_fail_percentage: 1
serial:

} install data on first 5% then 30% then another 50%

* playbook | apache.yml.
tasks:
-name:
when: ansible_distri == "Linux" → case-sensitive

→ [Fatces]

> ansible all -m Setup | grep Ubuntu → (will seach Ubuntu keyword in Facts)

"ansible_distribution" == "ubuntu"

* docker.yml } → to Setup env. ansible.

* docker module container

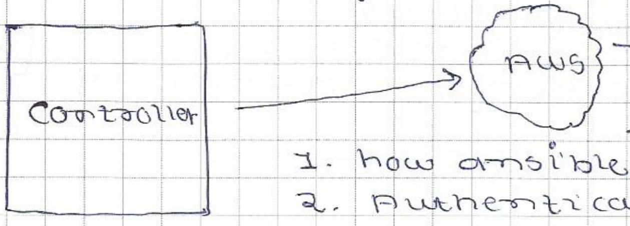
* ngix.yml →

tasks:
→ apt
—
— service
↳ state.

} Prod. Range config.

* Labs| ansible| playbook|. (Roles) [

> AWS Cli
> boto



→ Create an instance.
→ connectivity

1. how ansible connects to your aws account?
2. Authentication ?
↳ is Ravi allowed to make changes ?

* Ansible+ec2.txt
pip - python package manager.

* Pip install awscli (Aws command)
" " boto (check Aws installed)

> Aws --version.
aws-cli | 1.w

> Aws - Account
↓ how Aws authentcates ?

AwsCli >
Services> Security, Idendenty
↳ IAm (identity, access)
- access to team-memer or friends
→ Add Roles.

→ adding users & their Specific Roles.

> Add user → username = ravi

> Add user to group. (If any)
> Attach existing policies to Aws directly.

" Amazon EC2FullAccess"

* Create user (URL - COPY)

Ravi AccesskeyId key & Secret access key password

* Instances: 4 instances already + Add ~~man~~ one more

> aws configure → to Ansible
  Access Id: _
  Access key: _

> aws ec2 describe-instances --output table

> aws configure

* create user → configure (Aws-cli, boto)

* vi ec2.yml

(From Local machin)

```
- name: Create EC2 Instance
  hosts: Localhost — Local machine
  tasks:
    - ec2 (module)
       key-name: devopsravi
       instant_type: t2.micro
       image: image-id ( ami- 0SC1
       wait: yes
       group: # mysecgroup.
       count: 2
       vpc_subnet_id: subnet- 930
       assign_public_ip: yes
       region: us-east-2
    register: out
```

key file name (→)

[everything is installed on a local machine]

> ansible-playbook Aws.yml
  ec2 → last

* provision, config. manage, orchastration, Create, delete files (.

→ (step by step task execution)

* ansible with Local vm creation possible

* Aws → key Pairs → devopsravi (Rakesh).

* IaS(infrastructure as a code) → [write a code & Procure them)

* Instance creation & infrastructure creation

* Ansible cant do anything for user-Roles (LDAP) security people do the

* If [illegible]

* GDrive → playbooks (Run them in order)

* Ansible Roles:

Ad-Hoc : one Liner

Play-Book : set of Instructions → [Like shell scripting linux]

Roles: Playbook execution

→ gdrive

copy
abcd → Controller machine (abcd file) ⟵ abcd file should be available on controller machine]
/tmp → /tmp destination location

(git clone, → copy)

given you but No Resources given.

→ copy .war file
→ copy dependencies & all.

→ Completion package of ~~class~~ task executions (Ansible Roles)

* Package Complete - Ansible Roles. (Abcd file with executions)

> Roles - convert to Roles & share with Friends

Labs ... / ansible / Roles / ... (everything inside Roles)

Roles: Set up folders.

> Roles: Further Level abstractions (Very minimal playbook creations)

> ~~corrte~~ Files → put on default files Folder.

files: → where files located
meta: basic information
template.

{Roles is a set of folder} → make minimal conf. file (Rather than writing everything one file)

tasks: - write all your playbook contents
↳ what you want to execute.

vars: vars Folders. (all variables)

* tomcat → files - whatever you need
tasks - main.yml (install java)
- group
- Download tomcate
copy:
src: files:(files Folder)

→ / labs/ ansible / roles (how to execute them)
more tomcate-role.yml
roles:
- tomcat

> install java

Running Handler:

} using ansible, you can deploy docker container

| label ansible: # → variables
— files
— ~~m~~ tasks
—

> Role: finish in everything in a Role [write onces & call it 5 times]

> notify: restart tomcat. [5 times]

> ~~handler~~: Restart tomcate
nothing
{but a handler}

↳ handler (method execution)
- name: restart tomcat
service: name: tomcated
state = started

> ansible - galaxy: get pre-defined Roles. (d like github,

(mongdb instatlation

> ansible-galaxy ~~init~~ role init ravi

> ls -l > ~~ad hoc~~ files
~~install~~ tasks
~~playbooks~~

* template: execute playbook on 5-machions

ip1 Ravi	ip2 Ramkri	ip3 Vijay	ip4 Parmarth

(ginja-Concept.)

> templat. yml.
↑ src: config. j2 (ginja template) ← create this file

config. j2
name= {{ }} ↙ dynamic concepts

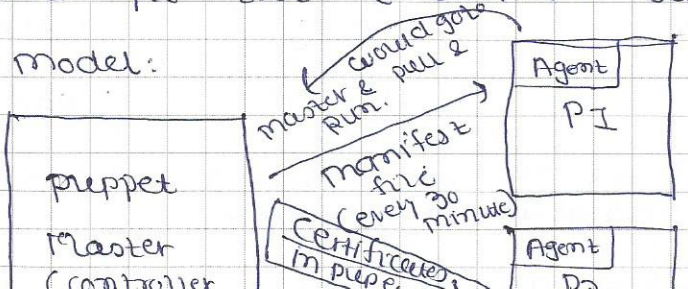
* Ansible- tower (UI based) [4gb min.]

*[SSL (Secure socket layer)] * Puppet * [Agent must Run, if not working, then no data fetch possible)

> Config. managle. tool. (different Ban brand)

> working model:

→ write all code in master { Manifest



puppet
Master
(controller
would goto master & pull & Run.
manifest file (every 30 minute)
Certificates in puppe
Agent
P1
Agent
P2

puppet Nodes or Agents. (execute)

puppet vs Ansible
Ruby = python