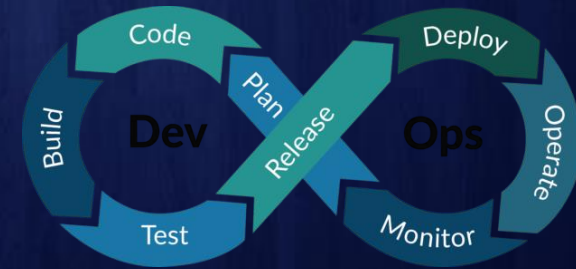


AWS SAA + SysOps + Developer + DevOps Course #Day-66

We will start at **8 AM**,
Stay tuned



RAKESH TANINKI

LEARN TO UNLEARN



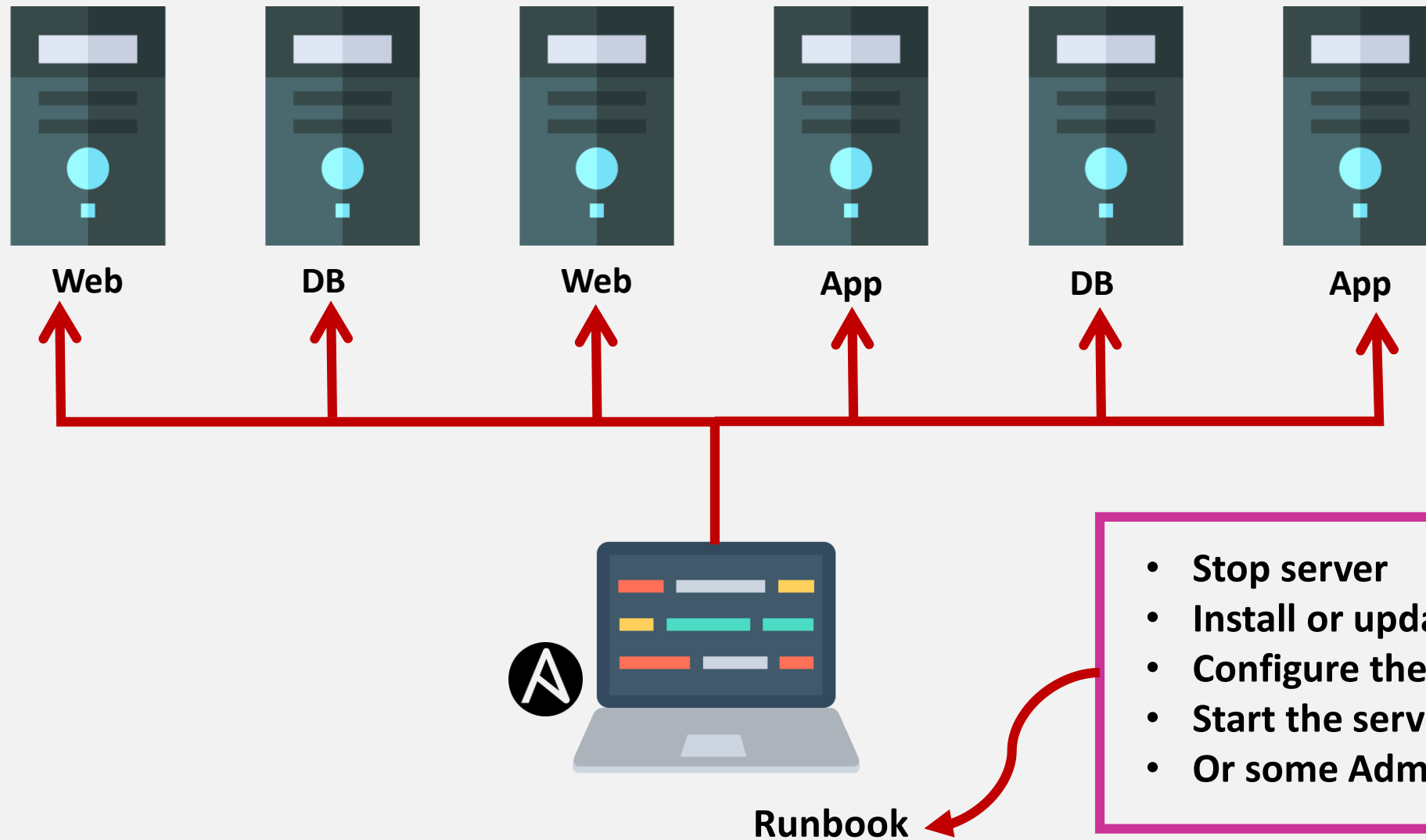
Today's topics:

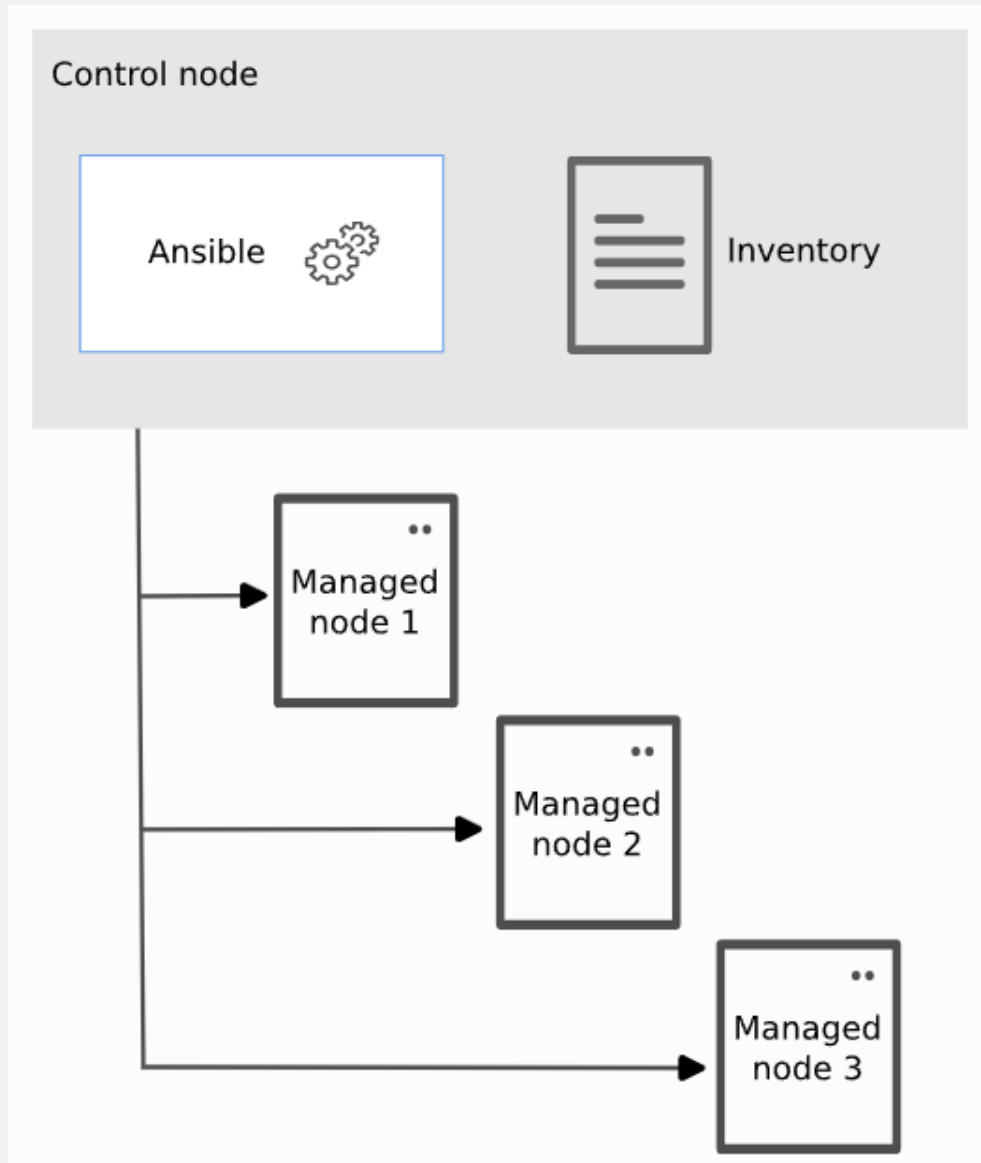


- **Ansible**
 - Introduction
 - Architecture
 - Installation
 - Inventory
 - Variables
 - Playbooks - basics

- Ansible is an open-source IT engine that **automates** application deployment, cloud provisioning, configuration management, and other IT tools.
- Simple
- Powerful
- Agentless
- It uses the simple playbooks – instead of large lines scripts like shell, etc

Ansible – Usecase





1. Control Node:

- Ansible installed system
- It controls all the managed servers

2. Managed Node:

- Remote system or host controlled by Ansible

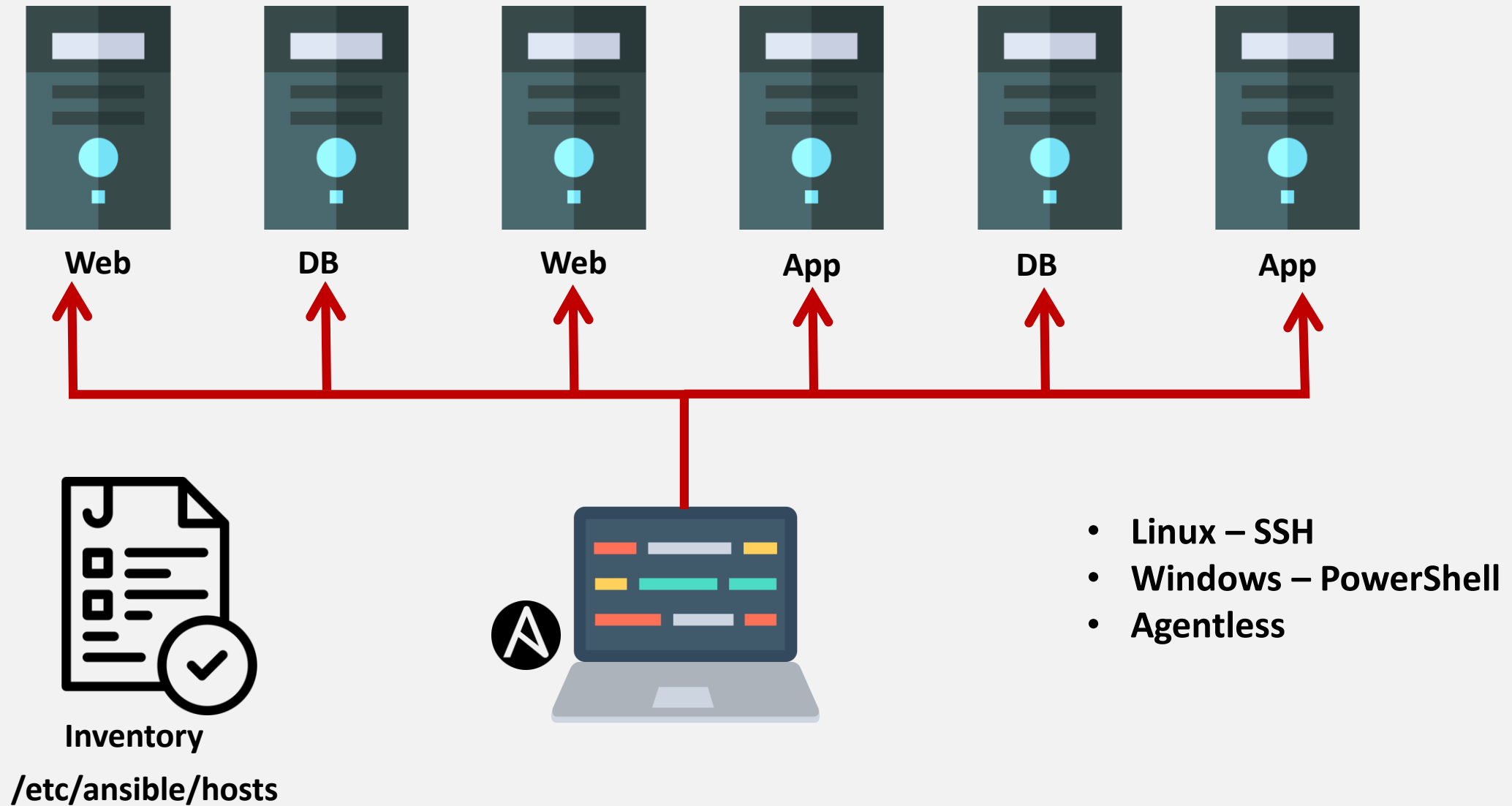
3. Inventory:

- Logical organization of managed nodes



- For Ansible installation Python is needed
- After installing python, run the below command
- *python3 -m pip install --user ansible*
- If pip not found
 - *curl https://bootstrap.pypa.io/get-pip.py -o get-pip.py*
 - *python3 get-pip.py --user*
 - *python3 -m pip install --user ansible*
- Run *ansible* or *ansible-inventory* to test the Ansible installation

Ansible – Inventory



Ansible inventory supports in ini and yaml formats

```
server1.company.com
```

```
server2.company.com
```

```
[db]
```

```
server3.company.com
```

```
1.2.3.4
```

```
[app]
```

```
11.22.33.44
```

```
server4.company.com
```



```
app ansible_host=server1.company.com  
db  ansible_host=server2.company.com  
web ansible_host=server3.company.com
```

- **ansible_connection** - ssh/winrm/localhost
- **ansible_port** – 22/5986
- **ansible_user** – Administrator / root
- **ansible_ssh_pass** – Password
- **ansible_host**

```
app ansible_host=server1.company.com ansible_connection=ssh
db  ansible_host=server2.company.com ansible_connection=rm
web ansible_host=server3.company.com ansible_connection=ssh

localhost ansible_connection=localhost
```

- **ansible_connection** - ssh/winrm/localhost
- **ansible_port** – 22/5986
- **ansible_user** – Administrator / root
- **ansible_ssh_pass** – Password

```
app ansible_host=server1.company.com ansible_connection=ssh ansible_user=root ansible_ssh_pass=!@#$!@#$
db ansible_host=server2.company.com ansible_connection=rm ansible_user=admin
web ansible_host=server3.company.com ansible_connection=ssh ansible_user=root

localhost ansible_connection=localhost
```

- **ansible_connection** - ssh/winrm/localhost
- **ansible_port** – 22/5986
- **ansible_user** – Administrator / root
- **ansible_ssh_pass** – Password
- **ansible_password**

```
hosts.yaml: ...  
all:  
  children:  
    webservers:  
      hosts:  
        server1.example.com  
        server2.example.com  
    dbservers:  
      hosts:  
        server3.example.com  
        server4.example.com
```

- Stores the values that varies with each host
- Types:
 - String:
 - username: “admin”
 - Number:
 - max_connections: 100
 - Boolean:
 - debug_mode: true / false
- Variables can read using jinja syntax – `{{var_name}}`

- **List:**
 - **packages:**
 - nginx
 - postgresql
 - git
- **For list – {{var_list[0]}}**
- **Dict:**
 - **user:**
 - name: “admin”
 - password: “secret”
- **For dict, {{user.name}}**



- Playbooks are for Ansible orchestration
- We define what tasks we want to perform the managed nodes

```
# complex ansible playbook
```

- Deploy VMs on to the public cloud
- Add storage to all vms
- Setup load balancing
- Setup Autoscaling
- Setup monitoring
- Update the domain

```
# simple ansible playbook
```

- Run commnad1 on server1
- Run command2 on server2
- Restart server1
- Restart server2



- **Playbooks are defined in YAML format, .yaml extensions**
 - **Play – defines a set of activities(tasks) to be run on hosts**
 - **Task – action performed on the host**
 - **Execute a command**
 - **Run a script**
 - **Install a package**
 - **Shutdown/ Restart**



Ansible – Playbook Sample

- Playbooks are defined in YAML format, .yaml extensions
- Play – defines a set of activities(tasks) to be run on hosts

```
ansible-playbook playbook.yml
```

```
-  
  name: play1  
  hosts: localhost  
  tasks:  
    - name: Exceute command of 'Date'  
      command: date  
  
    - name: Install httpd package  
      yum:  
        name: httpd  
        state: present  
  
    - name: Start httpd service  
      service:  
        name: httpd  
        state: started  
        enabled: yes
```



Thank you, will meet in tomorrow's session

