

## Ansible Notes

### Introduction to Ansible

- **Ansible** is an open-source IT automation tool used for configuration management, application deployment, and task automation.
- Operates without agents; it uses **SSH** or **WinRM** for communication.
- Simple to learn due to its **YAML-based** playbooks.
- Ensures **idempotency**, meaning running the same playbook multiple times will yield the same result.

### Key Components

1. **Control Node:** The machine where Ansible is installed and from which tasks are executed.
2. **Managed Nodes:** The target machines that Ansible automates.
3. **Inventory:** A file listing managed nodes (can be static or dynamic).
4. **Modules:** Pre-built units of work for system tasks (e.g., copy, service, yum).
5. **Playbooks:** YAML files that define the desired configuration or task.
6. **Roles:** A structured way to organize playbooks, variables, and files for reusability.
7. **Vault:** A feature to encrypt sensitive data like passwords or keys.

### Installation

#### On Control Node (Linux):

```
sudo apt update  
sudo apt install ansible -y
```

#### Verify Installation:

```
ansible --version
```

### Inventory File

- **Static Inventory:** Defined in `/etc/ansible/hosts`.

```
[webservers]  
192.168.1.10  
192.168.1.11 ansible_user=ubuntu ansible_ssh_private_key_file=~/.ssh/id_rsa
```

```
[databases]  
dbserver.example.com
```

- **Dynamic Inventory:** Generated dynamically using scripts or cloud integrations (e.g., AWS, GCP).

### Basic Commands

1. **Ping All Hosts:**

```
ansible all -m ping
```

2. **Run a Command:**

```
ansible webserver -m command -a "uptime"
```

3. **Copy Files:**

```
ansible all -m copy -a "src=/local/path dest=/remote/path"
```

4. **Install Packages:**

```
ansible all -m yum -a "name=httpd state=present"
```

## Writing a Playbook

### Example: Install and Start Nginx

```
---  
- name: Install and start Nginx  
  hosts: webserver  
  become: yes  
  
  tasks:  
    - name: Install Nginx  
      apt:  
        name: nginx  
        state: present  
  
    - name: Start Nginx service  
      service:  
        name: nginx  
        state: started
```

### Run the Playbook:

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

## Roles

### Directory Structure:

```
roles/  
  webserver/  
    tasks/  
      main.yml
```

```
handlers/  
  main.yml  
templates/  
  nginx.conf.j2  
files/  
  index.html  
vars/  
  main.yml
```

### Using a Role in a Playbook:

```
---  
- name: Deploy webserver  
  hosts: webserver  
  roles:  
    - webserver
```

### Ansible Vault

#### Encrypt a File:

```
ansible-vault encrypt secrets.yml
```

#### Decrypt a File:

```
ansible-vault decrypt secrets.yml
```

### Use Vault in a Playbook:

```
---  
- name: Secure playbook example  
  hosts: webserver  
  vars_files:  
    - secrets.yml  
  tasks:  
    - name: Print secret  
      debug:  
        msg: "{{ secret_key }}"
```

### Best Practices

1. Use **roles** for better organization.
2. Encrypt sensitive data using **Ansible Vault**.
3. Maintain a **consistent inventory structure**.
4. Test playbooks in staging environments before production.
5. Use **version control** (e.g., Git) to manage playbook versions.
6. Employ **idempotent modules** to ensure predictable outcomes.
7. Leverage **dynamic inventory** for cloud environments.

## Common Use Cases

1. **Application Deployment:** Automate deployments of web and database applications.
2. **Configuration Management:** Maintain consistent configurations across servers.
3. **Patch Management:** Apply security patches to systems.
4. **Cloud Automation:** Provision cloud resources (e.g., AWS, Azure, GCP).
5. **Container Orchestration:** Manage Docker containers and Kubernetes clusters.

## Debugging Tips

1. **Check Syntax:**

```
ansible-playbook playbook.yml --syntax-check
```

2. **Run in Check Mode:**

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

3. **Increase Verbosity:**

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

---

This guide provides a foundational overview of Ansible. Expand your knowledge by exploring more modules, dynamic inventories, and Ansible Galaxy roles!