

# Ansible: From Zero to Hero

---

AUTOMATE IT INFRASTRUCTURE WITH EASE

PRESENTED BY: TARIQ GHOURI



# What is Ansible?

---

Open-source IT automation tool

Agentless architecture

Uses YAML for configuration

Ideal for configuration management, application deployment, and task automation

# Core Components of Ansible

---

Inventory: List of managed nodes

Modules: Units of work (e.g., yum, copy)

Playbooks: YAML files defining automation tasks

Tasks: Individual actions within a playbook

Handlers: Triggered tasks upon changes

# Installing Ansible

---

On Ubuntu: `sudo apt update && sudo apt install ansible`

On CentOS: `sudo yum install epel-release && sudo yum install ansible`

# Setting Up SSH Access

---

Ensure passwordless SSH between control node and managed nodes

Generate SSH Key: `ssh-keygen`

Copy Key to Managed Node: `ssh-copy-id user@managed_node`

# Understanding Inventory Files

---

INI Format: `[webservers] server1 ansible_host=192.168.1.10`

YAML Format: `all -> hosts -> server1: ansible_host: 192.168.1.10`

# Ad-Hoc Commands

---

Check Connectivity: `ansible all -m ping`

Install Package: `ansible webservers -m yum -a 'name=httpd state=present'`

# Writing Your First Playbook

---

Playbook: Install and start Apache

Define tasks using YAML structure

Use become: yes to run tasks as root



# Roles in Ansible

---

Purpose: Organize playbooks into reusable components

Structure: roles/webserver/{tasks, handlers, templates, vars}

# Commonly Used Modules

---

yum: Manage packages on RHEL-based systems

apt: Manage packages on Debian-based systems

service: Manage services

copy: Copy files to remote machines

template: Deploy files from templates

user: Manage user accounts

# Testing and Debugging

---

Check Syntax: `ansible-playbook playbook.yml --syntax-check`

Dry Run: `ansible-playbook playbook.yml --check`

Verbose Output: `ansible-playbook playbook.yml -v`

# Real-World Use Cases

---

Automated server provisioning

Application deployment

CI/CD pipelines

Cloud infrastructure management

# Conclusion

---

Ansible simplifies IT automation

Agentless and easy to learn

Supports complex deployments with simple YAML syntax

Strong community and extensive resources