Ansible Deployment Documentation

Created By: Surya Prakash

Date: 27 July 2025

Purpose: Step-by-step guide for setting up DB servers and web servers using Ansible

# Table of Contents

1. 1. Inventory Setup
2. 2. SSH Configuration & Pinging Nodes
3. 3. MySQL Database Server Installation
4. 4. Apache Web Server & Website Deployment
5. 5. Verification & Troubleshooting

# 1. Inventory Setup

File: inventory.ini

[db\_servers]  
10.128.0.10 ansible\_user=ubuntu ansible\_ssh\_private\_key\_file=/home/ubuntu/gcp-key ansible\_python\_interpreter=/usr/bin/python3  
  
[app\_servers]  
10.128.0.9 ansible\_user=ubuntu ansible\_ssh\_private\_key\_file=/home/ubuntu/gcp-key ansible\_python\_interpreter=/usr/bin/python3  
  
[all\_nodes:children]  
db\_servers  
app\_servers

Key Notes:  
- Groups: db\_servers, app\_servers  
- Uses SSH key authentication (gcp-key)  
- Explicitly sets Python 3 interpreter

# 2. SSH Configuration & Pinging Nodes

## A. Configure Passwordless SSH (Optional)

# Generate SSH key (if not existing)  
ssh-keygen -t rsa -b 4096  
  
# Copy key to nodes  
ssh-copy-id -i ~/.ssh/gcp-key ubuntu@10.128.0.9  
ssh-copy-id -i ~/.ssh/gcp-key ubuntu@10.128.0.10

## B. Test Connectivity

# Ping all nodes  
ansible all\_nodes -m ping -i inventory.ini  
  
# Check only DB servers  
ansible db\_servers -m ping -i inventory.ini  
  
Expected Output:  
10.128.0.9 | SUCCESS => {"ping": "pong"}  
10.128.0.10 | SUCCESS => {"ping": "pong"}

# 3. MySQL Database Server Installation

Playbook: install\_mysql.yml

---  
- name: Install and secure MySQL  
 hosts: db\_servers  
 become: yes  
 vars:  
 mysql\_root\_password: "your\_secure\_password" # CHANGE THIS  
  
 tasks:  
 - name: Install MySQL Server  
 apt:  
 name: mysql-server  
 state: present  
 update\_cache: yes  
  
 - name: Start and enable MySQL  
 service:  
 name: mysql  
 state: started  
 enabled: yes  
  
 - name: Secure MySQL installation  
 shell: |  
 mysql -e "ALTER USER 'root'@'localhost' IDENTIFIED WITH mysql\_native\_password BY '{{ mysql\_root\_password }}';"  
 mysql -e "FLUSH PRIVILEGES;"  
 no\_log: true  
  
 - name: Verify MySQL access  
 mysql\_query:  
 login\_user: root  
 login\_password: "{{ mysql\_root\_password }}"  
 query: "SHOW DATABASES;"  
 register: db\_check  
 no\_log: true  
  
 - name: Show databases  
 debug:  
 var: db\_check.result

Run Playbook:  
ansible-playbook -i inventory.ini install\_mysql.yml  
  
Verification:  
ansible db\_servers -i inventory.ini -a "mysql -uroot -p{{ mysql\_root\_password }} -e 'SHOW DATABASES;'" --become

# 4. Apache Web Server & Website Deployment

Playbook: deploy\_webapp.yml

---  
- name: Deploy Apache and website  
 hosts: app\_servers  
 become: yes  
 vars:  
 website\_name: "kool\_form\_site"  
 download\_url: "https://www.tooplate.com/zip-templates/2136\_kool\_form\_pack.zip"  
  
 tasks:  
 - name: Install Apache  
 apt:  
 name: apache2  
 state: present  
  
 - name: Download website template  
 get\_url:  
 url: "{{ download\_url }}"  
 dest: "/tmp/website.zip"  
  
 - name: Unzip template  
 unarchive:  
 src: "/tmp/website.zip"  
 dest: "/var/www/{{ website\_name }}"  
 remote\_src: yes  
  
 - name: Configure Apache  
 copy:  
 content: |  
 <VirtualHost \*:80>  
 DocumentRoot /var/www/{{ website\_name }}  
 <Directory /var/www/{{ website\_name }}>  
 AllowOverride All  
 Require all granted  
 </Directory>  
 </VirtualHost>  
 dest: "/etc/apache2/sites-available/{{ website\_name }}.conf"  
  
 - name: Enable site  
 command: a2ensite {{ website\_name }}.conf  
 notify: Restart Apache  
  
 handlers:  
 - name: Restart Apache  
 service:  
 name: apache2  
 state: restarted

Run Playbook:  
ansible-playbook -i inventory.ini deploy\_webapp.yml  
  
Verify Website:  
curl http://10.128.0.9

# 5. Verification & Troubleshooting

Common Issues & Fixes:  
  
| Issue | Solution |  
|--------------------------|----------------------------------------------------|  
| SSH timeout | Check firewall (sudo ufw allow 22) |  
| MySQL root access denied | Re-run secure installation step |  
| Apache not serving files | Check folder permissions (chmod -R 755 /var/www) |  
  
Key Commands:  
# Check service status  
ansible app\_servers -a "systemctl status apache2" --become  
  
# View Apache logs  
ansible app\_servers -a "tail -n 20 /var/log/apache2/error.log" --become

# Conclusion

This document covered:  
✅ Ansible Inventory Setup  
✅ SSH Access Configuration  
✅ MySQL Installation on DB Servers  
✅ Apache Deployment + Static Website Hosting  
  
Next Steps:  
🔁 Automate backups (e.g. mysqldump via cron)  
📊 Add Monitoring (e.g. Prometheus + Grafana)  
🔒 Configure HTTPS (Let’s Encrypt + Certbot)  
  
Save this guide for future reference! 🚀