**Lesson 01 Demo 01**

**Setting Up LAMP Stack Using Ansible**

**Objective:** To set up and configure LAMP Stack dependencies with Ansible to pre-configure the infrastructure for new developers and automate the configuration management for the same

**Tools required:** Ubuntu OS

**Prerequisites:** You need to have Ansible installed.

Steps to be followed:

1. Configure the Ansible playbook
2. Execute the LAMP playbook

**Step 1: Configure the Ansible playbook**

1. Execute the command given below to check if Ansible is installed:

**ansible --version**

**A screenshot of a computer code

Description automatically generated**

1. Execute the following commands to navigate to the **hosts** directory:

**cd /etc/ansible**

**nano hosts**

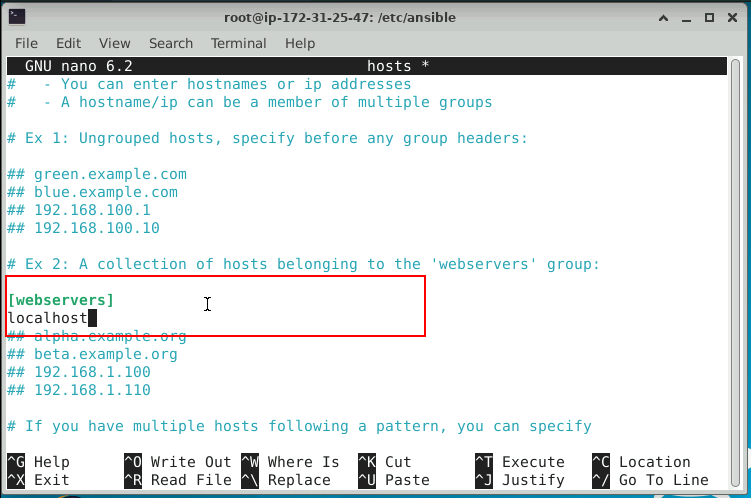


|  |
| --- |
| **Note:** You may need to use the commands with **sudo** to change into the root user. |

1. Configure the following **[webservers]** in the **hosts** file and save it:

**[webservers]**

**localhost**



1. Create an Ansible playbook with the command given below:

**mkdir AnsibleTest**

**cd AnsibleTest**

**nano lamp.yaml**

A screenshot of a computer

Description automatically generated

1. Enter the following YAML script in the **lamp.yaml** file to set up a LAMP stack (Linux, Apache, MySQL, PHP) on an Ubuntu server:

**---**

**- name: Install LAMP stack on Ubuntu**

**hosts: lamp\_server**

**become: yes**

**vars:**

**mysql\_root\_password: "your\_mysql\_root\_password"**

**php\_packages:**

**- php**

**- php-mysql**

**- libapache2-mod-php**

**- php-cli**

**- php-curl**

**- php-json**

**- php-mbstring**

**- php-xml**

**- php-zip**

**tasks:**

**- name: Update apt package index**

**apt:**

**update\_cache: yes**

**- name: Install Apache**

**apt:**

**name: apache2**

**state: present**

**- name: Start and enable Apache service**

**service:**

**name: apache2**

**state: started**

**enabled: yes**

**- name: Install MySQL**

**debconf:**

**name: "mysql-server"**

**question: "mysql-server/root\_password"**

**value: "{{ mysql\_root\_password }}"**

**vtype: "password"**

**- name: Install MySQL again to set the root password**

**debconf:**

**name: "mysql-server"**

**question: "mysql-server/root\_password\_again"**

**value: "{{ mysql\_root\_password }}"**

**vtype: "password"**

**- name: Install MySQL server**

**apt:**

**name: mysql-server**

**state: present**

**- name: Start and enable MySQL service**

**service:**

**name: mysql**

**state: started**

**enabled: yes**

**- name: Install PHP and related packages**

**apt:**

**name: "{{ php\_packages }}"**

**state: present**

**- name: Create info.php to test PHP**

**copy:**

**dest: /var/www/html/info.php**

**content: |**

**<?php**

**phpinfo();**

**?>**

**- name: Restart Apache to apply PHP installation**

**service:**

**name: apache2**

**state: restarted**

**handlers:**

**- name: Restart Apache**

**service:**

**name: apache2**

**state: restarted**

A screenshot of a computer

Description automatically generated

**Step 2: Execute the LAMP playbook**

* 1. Execute the following command to run the playbook:  
     **ansible-playbook lamp.yaml**

A screenshot of a computer

Description automatically generated

By following these steps, you have successfully set up and configured LAMP Stack dependencies with Ansible to pre-configure the infrastructure for new developers and automate the configuration management.

|  |
| --- |
| **Note:** In the real-world scenario, you will have a list of hosts where you will do the configuration management. This will be discussed in the upcoming lessons. |