Setting Up Inventory Files to Manage Groups of Hosts

Table of Contents

1. Introduction
2. Problem Statement
3. Prerequisites
    - Software Requirements
4. Step-by-Step Guide to Setting Up Inventory Files
    - Step 1: Create an Inventory File
    - Step 2: Define Host Groups
    - Step 3: Use the Inventory in a Playbook
5. Verifying the Setup
6. Supported References

Introduction

Inventory files in Ansible are used to define the hosts and groups of hosts that you will manage with Ansible. This allows you to organize your infrastructure logically, making it easier to run playbooks against specific sets of servers.

> **Note: What is an Inventory File?**
>
> ```
> An inventory file is a configuration file that lists the nodes or servers
> that Ansible can manage. It can be in INI, YAML, or JSON format and can
> include variables and groupings for easier management.
> ```

In this guide, we'll explore how to set up inventory files to manage groups of hosts effectively.

Problem Statement

Managing a large number of servers can be challenging without proper organization. Using inventory files, you can group servers based on their roles, environments, or any other criteria. This simplifies running Ansible playbooks and tasks across multiple servers.

Prerequisites

Completion of all previous lab guides (up to Lab Guide-07) is required before proceeding with Lab Guide-08.

Software Requirements

- **Ansible 2.9+**: Installed on your control node (WSL for Windows users).
- **WSL (Windows Subsystem for Linux)**: If using Windows as your control node.

Step-by-Step Guide to Setting Up Inventory Files

Step 1: Create an Inventory File

1. **Create an Inventory File**:
    - Create a new file named `inventory.ini` (you can choose a different name as well):

```
nano inventory.ini
```

2. **Add Basic Host Entries**:
    - In the inventory file, add the IP addresses or hostnames of your servers:

```
[windows]
192.168.0.102

[database_servers]
192.168.0.201
192.168.0.202
```

 images

This file defines two groups: `windows` and `database_servers`.

Step 2: Define Host Groups

You can further define variables specific to each group in the same inventory file(`inventory.ini`).

1. **Add Variables for Groups**:
    - You can specify variables that apply to all hosts in a group:

```
[windows]
192.168.0.102 ansible_user=ansible_user ansible_password=P@ssw0rd

[database_servers]
192.168.0.201 ansible_user=ansible_user ansible_password=P@ssw0rd
192.168.0.202 ansible_user=ansible_user ansible_password=P@ssw0rd

[windows:vars]
ansible_connection=winrm
ansible_winrm_transport=basic
ansible_winrm_server_cert_validation=ignore
ansible_port=5985

[database_servers:vars]
db_port=5432
```

Here, `web_port` and `db_port` are example variables defined for the respective groups.

Step 3: Use the Inventory in a Playbook

1. **Create a Playbook to Use the Inventory**:
   - Create a new playbook named `deploy_web_servers.yml`:

```
nano deploy_web_servers.yml
```

2. **Add Playbook Content**:
   - Use the inventory file to define which hosts to target:

```
---
- name: Deploy Web Server
  hosts: windows
  tasks:
    - name: Ensure Apache is installed
      win_feature:
        name: Web-Server
        state: present
```

This playbook targets the hosts defined under the `windows` group in the inventory file.

Verifying the Setup

Once you have defined your inventory and playbook, you can verify your setup by running a simple Ansible command to ping winodws as the host:

```
ansible windows -i inventory.ini -m win_ping
```

You should receive a `pong` response from each of your managed hosts if everything is set up correctly.

Supported References

- Ansible Inventory Documentation
- Ansible Playbook Documentation
