Install and configure Ansible on a control node.

Table of Contents

Introduction

This guide will walk you through installing and configuring Ansible on a Windows system using WSL (Windows Subsystem for Linux). Ansible allows you to manage remote machines, automate tasks, and deploy applications across multiple servers, making infrastructure management simpler and more efficient.

Problem Statement

Setting up Ansible on a Windows system can be challenging due to its native Linux-based environment. This document explains how to install and configure Ansible on Windows using WSL, allowing Windows users to utilize Ansible's automation capabilities seamlessly.

Prerequisites

Software Required

- **Windows 10 or later**
- **Windows Subsystem for Linux (WSL)**
- **Python 3.8 or later**
- **Ansible 2.9 or later**

Hardware Requirement

- **4 GB RAM** or higher
- **20 GB** of free disk space
- **Internet access** for downloading packages

Implementation Steps

Step-1: Install Windows Subsystem for Linux (WSL)

To run Ansible on a Windows machine, you need to install WSL to provide a Linux environment. Follow the steps below to enable and set up WSL.

1. **Enable WSL**

   ```
   wsl --install
   ```

   Once WSL is installed, restart your Windows system.

   images

2. **Install Linux Distribution**

   Install Ubuntu as your Linux distribution:

   ```
   wsl --install -d Ubuntu
   ```

   > **Note**: During the setup, you will be asked to create a username and password for your WSL environment.

   images

3. **Update Linux Packages**

   After setting up Ubuntu, update and upgrade the packages:

   ```
   sudo apt update
   sudo apt upgrade
   ```

   images images

Step-2: Install Python and PIP

Ansible requires Python to function, so you need to install both Python and PIP on your WSL environment.

1. Install Python:

   ```
   sudo apt install python3
   ```

   images

2. Install PIP:

   ```
   sudo apt install python3-pip
   ```

   images

3. Verify installation:

   ```
   python3 --version
   pip3 --version
   ```

   images

Step-3: Install Ansible

After setting up Python, you can install Ansible using the following commands.

1. Add the Ansible repository:

   ```
   sudo apt-add-repository --yes --update ppa:ansible/ansible
   ```

   images

2. Install Ansible:

   ```
   sudo apt install ansible
   ```

   images

3. Verify the installation:

   ```
   ansible --version
   ```

   images

Step-4: Create a Windows User and Configure WinRM

Before proceeding with Ansible configuration, ensure WinRM is configured on your Windows system for remote management.

1. **Create a new user in Windows**:

    - Username: `ansible_user`
    - Password: `P@ssw0rd`
    - Ensure the account type is set to **Administrator**.

2. **Open PowerShell with Administrator access** and check if WinRM is running. If it is not running, start the service with the following commands:

   ```
   winrm quickconfig
   ```

3. **Set WinRM configuration**:

   Configure WinRM to allow basic authentication:

   ```
   winrm set winrm/config/service/auth '@{Basic="true"}'
   ```

Step-5: Configure Ansible

1. **Edit the Ansible Inventory**:

   In your WSL environment, open the Ansible inventory file:

   ```
   sudo nano /etc/ansible/hosts
   ```

2. **Add Windows host configuration**:

   Add the following configuration to the file, replacing `<ip-address>` with the IP of the Windows machine you want to manage:

   ```
   [windows]
   <ip-address>

   [windows:vars]
   ansible_user=ansible_user
   ansible_password=P@ssw0rd
   ansible_connection=winrm
   ansible_winrm_transport=basic
   ansible_winrm_server_cert_validation=ignore
   ansible_port=5985
   ```

   images

Step-6: Test Ansible Installation

1. **Ping remote Windows hosts**:

   Test Ansible to ensure that it can communicate with the Windows host by running:

   ```
   ansible windows -m win_ping
   ```

   images

   If successful, you will see a response confirming that the connection is established.

Supported Reference

For more detailed references on Ansible for Windows management, you can refer to:

- Ansible Documentation
- WSL Documentation
- Pywinrm Documentation
