How to Install Ansible?

While installing Ansible, it's essential to have two machines. Let's name our first machine, as 'Server' that will act as our managed node, and the second machine named 'Node' that will act as the controller node.

Now, let's get started on this and understand the Ansible installation process in detail.

Step 1: Update Your Control Node

Before installing any new software, it is important to ensure that your existing operating version is up-to-date. Enter the command mentioned below to start your task.

yum update

Step 2: Install EPEL Repository

Moving on, install the EPEL repository on the system.

yum install epel-release

EPEL provides easy access to install commonly used packages on CentOS.

Step 3: Install Ansible

The next step is to install the Ansible package from the EPEL repository.

yum install ansible

Step 4: Create a User for Ansible

Let's create a non-root user on both the nodes that will run our Ansible playbooks. In this demo, we will use "Simplilearn"' as the username (but any username can be added). Also, you should ensure that you use the same username on both the nodes (i.e., controller node and your managed node).

Add a user and set a password onto your Controller node.

useradd simplilearn

passwd Simpli@123

Add the admin user, and set a password onto your Managed node.

useradd simplilearn

passwd Simpli@123

Step 5: Configure Our Admin User for SSH Access

Configure the admin user (i.e. Simplilearn user) so that it can access the Managed node without a password. Then, ensure that you set up an SSH key pair to the Simplilearn user.

Now, run the following command (in the control node) to generate an SSH key pair.

ssh-keygen

Then, copy the public key and paste it to our Managed node with the command below.

ssh-copy-id node.kb.liquidweb.com

Step 6: Create an Inventory

An inventory list is created to identify your managed nodes.

Log in to your control node as the admin user to connect the Managed node to the inventory.

vim /home/admin/inventory

Now, enter 'i' to get to the insert mode, and add the hostname of the Managed Node. node.kb.liquidweb.com

Now, save the file by typing '[ESC]:wq'.

Step 7: Create an Ansible Playbook

Here, we will create a simple Ansible playbook by installing Nginx on the Managed Node.

First, log onto your Controller Node as the "Simplilearn" user and create a file with a descriptive name.
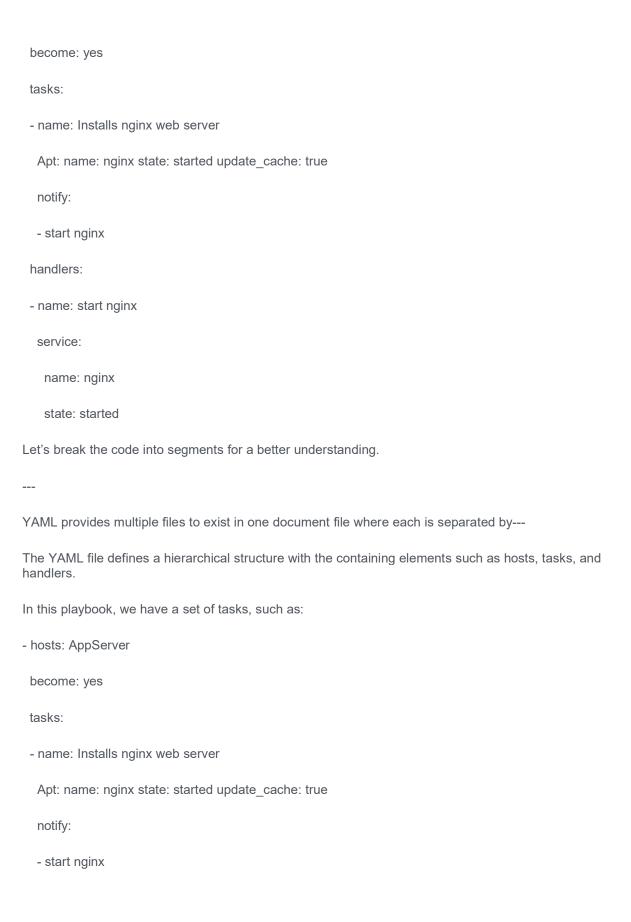
vim /home/simplilearn/install-nginx.yml

These playbooks are written in YAML language which is human-readable (as shown below).

Now, go to the insert mode and add the following text to your playbook.

--

- hosts: AppServer

```
    become: yes

  tasks:

  - name: Installs nginx web server

    Apt: name: nginx state: started update_cache: true

    notify:

    - start nginx

  handlers:

  - name: start nginx

    service:

      name: nginx

      state: started
```

Let's break the code into segments for a better understanding.

YAML provides multiple files to exist in one document file where each is separated by---

The YAML file defines a hierarchical structure with the containing elements such as hosts, tasks, and handlers.

In this playbook, we have a set of tasks, such as:

```
- hosts: AppServer

  become: yes

  tasks:

  - name: Installs nginx web server

    Apt: name: nginx state: started update_cache: true

    notify:

    - start nginx
```

Task performs all the major operations in the file.

Here, 'notify' consists of a list with one item, which is called "start nginx." Notify is not an internal command of Ansible but a reference to a handler that is responsible for performing a function when it is called by a task.

Handlers are the same as hosts and tasks, but they operate only when instructed by a task on the client system.

```
 handlers:

  - name: start nginx

    service:

      name: nginx

      state: started
```

The code above defines the "start nginx" handler. The handler will execute after the nginx web server is installed. We can save this playbook into a file called something like "nginx.yml".

Then type '[ESC]:wq' to save this and exit.

Note: Prioritize using only spaces and not tabs. Make sure to use consistent spacing for your YAML file to avoid errors.

Step 8: Run the Playbook

Our Ansible playbook is built. Now, to run the playbook, type the following command on the controller node:

ansible-playbook -i /home/admin/inventory /home/admin/install-nginx.yml

In the command above, we have added the inventory file with the "-i" option, followed by the playbook path.