How Ansible Works?

There are many similar automation tools available like Puppet, Capistrano, Chef, Salt, Space Walk etc, but Ansible categorize into two types of server: controlling machines and nodes.

The controlling machine, where Ansible is installed and Nodes are managed by this controlling machine over SSH. The location of nodes are specified by controlling machine through its inventory.

The controlling machine (Ansible) deploys modules to nodes using SSH protocol and these modules are stored temporarily on remote nodes and communicate with the Ansible machine through a JSON connection over the standard output.

Ansible is agent-less, that means no need of any agent installation on remote nodes, so it means there are no any background daemons or programs are executing for Ansible, when it’s not managing any nodes.

Ansible can handle 100’s of nodes from a single system over SSH connection and the entire operation can be handled and executed by one single command ‘ansible’. But, in some cases, where you required to execute multiple commands for a deployment, here we can build playbooks.

Playbooks are bunch of commands which can perform multiple tasks and each playbooks are in YAML file format.

What’s the Use of Ansible

Ansible can be used in IT infrastructure to manage and deploy software applications to remote nodes. For example, let’s say you need to deploy a single software or multiple software to 100’s of nodes by a single command, here ansible comes into picture, with the help of Ansible you can deploy as many as applications to many nodes with one single command, but you must have a little programming knowledge for understanding the ansible scripts.

We’ve compiled a series on Ansible, title ‘Preparation for the Deployment of your IT Infrastructure with Ansible IT Automation Tool‘, through parts 1-4 and covers the following topics.

Part 1: How to Install and Configure Ansible for IT Management in Linux

Part 2: How to Use Anisble Playbooks to Automate Complex Tasks on Multiple Remote Servers

Part 3: How to Automate Simultaneous WordPress Deployments in Multiple Linux Servers Using Ansible

Part 4: Managing Encrypted YAMAL data with Ansible-Vault

In this article, we will show you how to install ‘Ansible’ on RHEL/CentOS 7/6, Fedora 21-19, Ubuntu 14.10-13.04 and Debian 7/6 systems and also we will go through some basics on how how to manage a server by installing packages, applying updates and much more from basic to pro.

Prerequisites

1. Operating System: RHEL/CentOS/Fedora and Ubuntu/Debian/Linux Mint
2. Jinja2: A modern, fast and easy to use stand-alone template engine for Python.
3. PyYAML: A YAML parser and emitter for the Python programming language.
4. parmiko: A native Python SSHv2 channel library.
5. httplib2: A comprehensive HTTP client library.
6. sshpass: A non-interactive ssh password authentication.

My Environment Setup

Controlling Machine – Ansible

Operating System : Linux Mint 17.1 Rebecca

IP Address : 192.168.0.254

Host-name : tecmint.instrcutor.com

User : tecmint

Remote Nodes

Node 1: 192.168.0.112

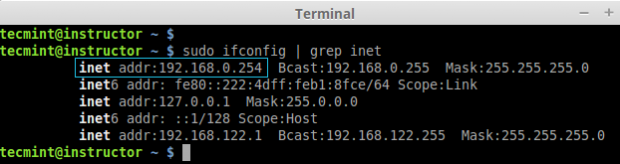
Node 2: 192.168.0.113

Node 3: 192.168.0.114

Step 1: Installing Controlling Machine – Ansible

1. Before installing ‘Ansible‘ on the server, let’s first verify the details of the server like hostname and IP Address. Login into server as a root user and execute the below command to confirm system settings that we’re going to use for this setup.

# sudo ifconfig | grep inet

[](https://www.tecmint.com/wp-content/uploads/2015/01/Verify-System-Details.png)

*Verify System Details*

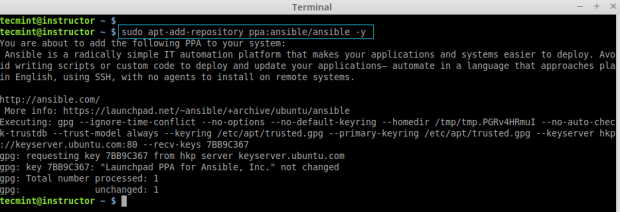
2. Once you confirm your system settings, it’s time to install ‘Ansible’ software on the system.

On Ubuntu/Debian/Linux Mint

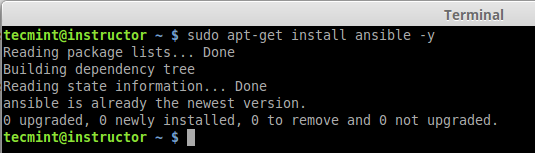
Here we are going to use official Ansible PPA repository on the system, just run the below commands to add the repository.

$ sudo apt-add-repository ppa:ansible/ansible -y

$ sudo apt-get update && sudo apt-get install ansible -y

[](https://www.tecmint.com/wp-content/uploads/2015/01/Add-Ansible-PPA3.png)

*Add Ansible PPA*

[](https://www.tecmint.com/wp-content/uploads/2015/01/Install-Ansible-in-Ubuntu1.png)

*Install Ansible in Ubuntu*

On RHEL/CentOS/Fedora

Unfortunately, there are no official Ansible repository for RedHat based clones, but we can install Ansible by enabling epel repository under RHEL/CentOS 6, 7 and currently supported fedora distributions.

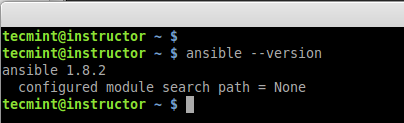
Fedora users can directly install Ansible through default repository, but if you are using RHEL/CentOS 6, 7, you have to enable EPEL repo.

After configuring epel repository, you can install Ansible using following command.

$ sudo yum install ansible -y

After installed successfully, you can verify the version by executing below command.

# ansible --version

[](https://www.tecmint.com/wp-content/uploads/2015/01/Verify-Ansible-Version1.png)

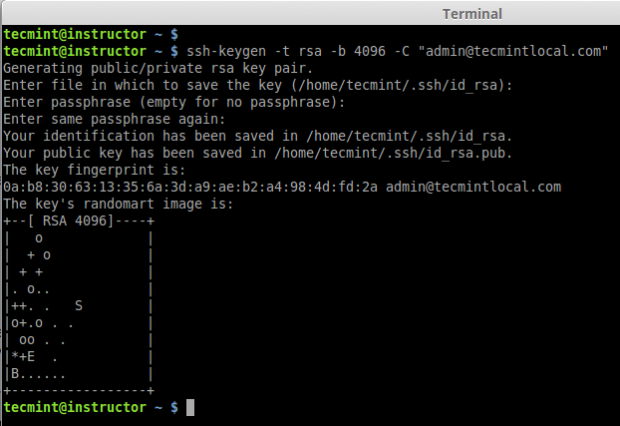
*Verify Ansible Version*

Step 2: Preparing SSH Keys to Remote Hosts

4. To perform any deployment or management from the localhost to remote host first we need to create and copy the ssh keys to the remote host. In every remote host there will be a user account tecmint (in your case may be different user).

First let we create a SSH key using below command and copy the key to remote hosts.

# ssh-keygen -t rsa -b 4096 -C "admin@tecmintlocal.com"

[](https://www.tecmint.com/wp-content/uploads/2015/01/Create-SSH-Key1.png)

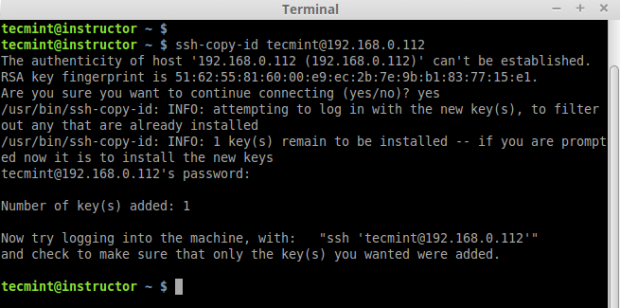
*Create SSH Key*

5. After creating SSH Key successfully, now copy the created key to all three remote server’s.

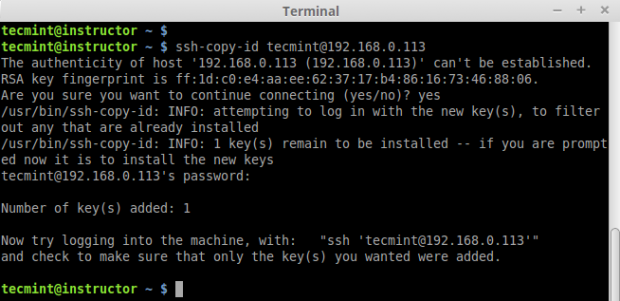
# ssh-copy-id tecmint@192.168.0.112

# ssh-copy-id tecmint@192.168.0.113

# ssh-copy-id tecmint@192.168.0.114

[](https://www.tecmint.com/wp-content/uploads/2015/01/Copy-SSH-Key1.png)

*Copy SSH Key Remote Server*

[](https://www.tecmint.com/wp-content/uploads/2015/01/Copy-SSH-Keys1.png)

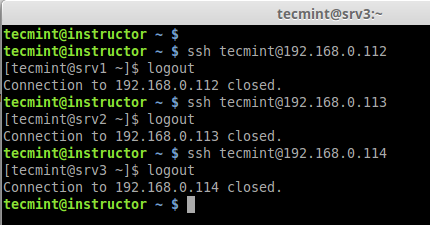
*Copy SSH Key Second Remote Host*

6. After copying all SSH Keys to remote host, now perform a ssh key authentication on all remote hosts to check whether authentication working or not.

$ ssh tecmint@192.168.0.112

$ ssh tecmint@192.168.0.113

$ ssh tecmint@192.168.0.114

[](https://www.tecmint.com/wp-content/uploads/2015/01/SSH-Key-Authentication1.png)

*SSH Key Authentication*