



What is Ansible, How Ansible works – Ansible Tutorials

📅 October 13, 2017

This tutorial describes you about the Ansible Introduction, What is Ansible, Why do we use Ansible? How Ansible works, Ansible Advantages, Ansible Terminologies and Ansible Installation. This will be very useful to the beginners who are interested to learn the Ansible from basics.

Ansible related articles something you have missed!

1. [What is Ansible, How Ansible works, Ansible Introduction, Ansible Basic Tutorials](#)
2. [Ansible Inventory Introduction – Ansible Beginner Tutorials](#)
3. [Ansible Ad hoc Commands – Ansible Tutorial for Beginners](#)
4. [Managing Ansible Configuration Files Explained with Examples](#)
5. [Understanding Ansible Playbook – Write your First Playbook](#)
6. [Ansible Roles Explained with Examples – Ansible Tutorials](#)
7. [How to use Ansible Vault to Protect Ansible Playbooks](#)

Lets get started.



What is Ansible? Ansible Introduction:

Ansible is an open source automation platform. Ansible can help you with configuration management, application deployment, task automation and also IT orchestration. It is very simple to setup, efficient and more powerful.

Ansible is available for free and runs on Linux, Mac or BSD. Aside from the free offering, Ansible also has an enterprise product called Ansible Tower.

Why do we use Ansible? or Advantages of Ansible:

We have lot of automations tools available in the market namely puppet, chef and saltstack. Obviously, each tools has its own features and functionality.

1. Ansible is free and open source.
2. Agentless. Unlike other tools, ansible doesnt require a master client model. which means, no agent to be installed at the client.
3. So how does it work? Ansible uses SSH for the operations on the existing network.
4. System requirement is very low.
5. Devloped in Python. Python is that it is inbuilt into most Unix and Linux deployments nowadays, so getting up and running is quicker.
6. lightweight and quick deployment. Ofcourse its lightweight and its very fast to deployment, becuae it uses the SSH, doesnt have master client communication compare to other tools.
7. Ansible uses YAML Syntax in configuration files.
8. Larger community. Ansible has a large and engaged community of users who can help answer your questions.

ALSO WATCH THIS "ANSIBLE" TUTORIAL VIDEO FREE ON OUR YOUTUBE CHANNEL

Ansible Tutorials | What is Ansible, Why do we use Ansible, ...



Ansible Terminologies

Before playing with ansible, we should have knowledge of its terminologies to understand the functionality.

Playbooks

Playbooks are expressed in YAML format and have a minimum of syntax, Playbooks are the language by which Ansible configures, administers, deploys systems, automate tasks and orchestrates.

Task

Playbooks exist to run tasks. Tasks combine an action with a name and optionally some other keywords (like looping directives). Handlers are also tasks, but they are a special kind of task that do not run unless they are notified by name when a task reports an underlying change on a remote system.

Tasks

A list of Task.

Action

An action is a part of a task that specifies which of the modules to run and which arguments to pass to that module. Each task can have only one action, but it may also have other parameters.

Facts

Facts are simply things that are discovered about remote nodes. While they can be used in playbooks and templates just like variables, facts are things that are inferred, rather than set. Facts are automatically discovered by Ansible when running plays by executing the internal setup module on the remote nodes. You never have to call the setup module explicitly, it just runs, but it can be disabled to save time if it is not needed or you can tell ansible to collect only a subset of the full facts via the `gather_subset`: option. For the convenience of users who are switching from other configuration management systems, the fact module will also pull in facts from the `ohai` and `facter` tools if they are installed. These are fact libraries from Chef and Puppet, respectively. (These may also be disabled via `gather_subset`.)

Host

A host is simply a remote machine that Ansible manages. They can have individual variables assigned to them, and can also be organized in groups. All hosts have a name they can be reached at (which is either an IP address or a domain name) and, optionally, a port number, if they are not to be accessed on the default SSH port.

Group

A group consists of several hosts assigned to a pool that can be conveniently targeted together, as well as given variables that they share in common.

Handlers

Handlers are just like regular tasks in an Ansible playbook but are only run if the Task contains a `notify` directive and also indicates that it changed something. For example, if a config file is changed, then the task referencing the config file templating operation may notify a service restart handler. This means services can be bounced only if they need to be restarted. Handlers can be used for things other than service restarts, but service restarts are the most common usage.

Ad Hoc

Refers to running Ansible to perform some quick command, using `/usr/bin/ansible`, rather than the orchestration language, which is `/usr/bin/ansible-playbook`. An example of an ad hoc command might be rebooting 50 machines in your infrastructure. Anything you can do ad hoc can be accomplished by writing a playbook and playbooks can also glue lots of other operations together.

Inventory

A file (by default, Ansible uses a simple INI format) that describes Hosts and Groups in Ansible. Inventory can also be provided via an Inventory Script (sometimes called an "External Inventory Script"). So you can define all your hosts in a single file as a Inventory either in the hostname format, ip address or fqdn.

Modules

Modules are the units of work that Ansible ships out to remote machines. Modules are kicked off by either `/usr/bin/ansible` or `/usr/bin/ansible-playbook` (where multiple tasks use lots of different modules in conjunction). Modules can be implemented in any language, including Perl, Bash, or Ruby – but can leverage some useful communal library code if written in Python. Modules just have to return JSON. Once modules are executed on remote machines, they are removed, so no long running daemons are used. Ansible refers to the collection of available modules as a library.

Library

A collection of modules made available to `/usr/bin/ansible` or an Ansible playbook.

Notify

The act of a task registering a change event and informing a handler task that another action needs to be run at the end of the play. If a handler is notified by multiple tasks, it will still be run only once. Handlers are run in the order they are listed, not in the order that they are notified.

Orchestration

Many software automation systems use this word to mean different things. Ansible uses it as a conductor would conduct an orchestra. A datacenter or cloud architecture is full of many systems, playing many parts – web servers, database servers, maybe load balancers, monitoring systems, continuous integration systems, etc. In performing any process, it is necessary to touch systems in particular orders, often to simulate rolling updates or to deploy software correctly. Some system may perform some steps, then others, then previous systems already processed may need to perform more steps. Along the way, emails may need to be sent or web services contacted. Ansible orchestration is all about modeling that kind of process.

Push Mode

Push mode is the default mode of Ansible. In fact, it's not really a mode at all – it's just how Ansible works when you aren't thinking about it. Push mode allows Ansible to be fine-grained and conduct nodes through complex orchestration processes without waiting for them to check in.

Pull Mode

By default, Ansible runs in push mode, which allows it very fine-grained control over when it talks to each system. Pull mode is provided for when you would rather have nodes check in every N minutes on a particular schedule. It uses a program called `ansible-pull` and can also be set up (or reconfigured) using a push-mode playbook. Most Ansible users use push mode, but pull mode is included for variety and the sake of having choices.

Roles

Roles are units of organization in Ansible. Assigning a role to a group of hosts (or a set of groups, or host patterns, etc.) implies that they should implement a specific behavior. A role may include applying certain variable values, certain tasks, and certain handlers – or just one or more of these things. Because of the file structure associated with a role, roles become redistributable units that allow you to share behavior among playbooks – or even

with other users.

Templates

Ansible can easily transfer files to remote systems but often it is desirable to substitute variables in other files. Variables may come from the inventory file, Host Vars, Group Vars, or Facts. Templates use the Jinja2 template engine and can also include logical constructs like loops and if statements.

When(Optional)

An optional conditional statement attached to a task that is used to determine if the task should run or not. If the expression following the when: keyword evaluates to false, the task will be ignored.

YAML

Ansible does not want to force people to write programming language code to automate infrastructure, so Ansible uses YAML to define playbook configuration languages and also variable files. YAML is nice because it has a minimum of syntax and is very clean and easy for people to skim. It is a good data format for configuration files and humans, but also machine readable. Ansible's usage of YAML stemmed from Michael DeHaan's first use of it inside of Cobbler around 2006. YAML is fairly popular in the dynamic language community and the format has libraries available for serialization in many languages (Python, Perl, Ruby, etc.).

These are the terminologies will be used in ansible. And you will be able to understand more on each things when you are playing with ansible.

Install Ansible in RHEL / CentOS / Debian Based Servers

If you are using a debian ubuntu systems, you can use apt-get to install ansible.

```
apt-get install ansible
```

If you are using rhel or centos, Enable the extras repos and use yum package manager to install ansible.

```
yum install ansible
```

Going forward, we will play more with ansible tool with some test cases. [Ansible](#), [What is Ansible](#), [How Ansible works](#), [Ansible Tutorial](#), [Ansible tutorial for beginners](#), [ansible playbook](#), [ansible installation](#), [Ansible basics](#), [ansible introduction](#), [ansible fundamentals](#), [introduction to ansible](#), [understanding ansible](#), [ansible learning](#), [how ansible works linux](#), [ansible master configuration management](#), [ansible tool in linux](#), [ansible tutorial linux pdf](#), [what is ansible linux](#), [ansible open source](#), [ansible wiki](#), [ansible openstack](#), [openstack ansible](#), [ansible complete tutorial](#), [ansible beginner tutorial](#)

More topics will be added soon.

Support Us: Share with your friends and groups.

Stay connected with us on social networking sites, Thank you.

[YouTube](#) | [Facebook](#) | [Twitter](#) | [Pinterest](#) | [Rss](#)

[ansible tutorial for beginners](#), [ansible tutorial online](#), [ansible training online](#), [ansible learning guide](#), [ansible quick tutorial](#), [ansible server tutorial](#), [ansible](#)

tool tutorial, ansible tutorial configuration management, ansible tutorial in linux, ansible tutorial step by step, ansible tutorial with examples, centos 6 ansible tutorial, centos 7 ansible tutorial, how to install ansible, ansible architecture tutorial, ansible automation tutorial, ansible automation tutorial pdf, ansible dashboard tutorial, ansible deploy tutorial, ansible devops tutorial, ansible docker tutorial, ansible e learning, ansible easy tutorial, ansible tutorial online, ansible tutorial open source, ansible tutorial pdf, ansible tutorial pdf download, ansible tutorial ppt, ansible tutorial ansiblelabs, ansible tutorial python, ansible tutorial redhat, ansible tutorial rhel, ansible tutorial site.pp, ansible tutorial ssh, ansible tutorial template, ansible tutorial video linux, ansible tutorial videos, ansible tutorial vm, ansible tutorial youtube, ansible ubuntu 12.04 tutorial, ansible users tutorial, unix ansible tutorial, using ansible tutorial, yoshinon ansible tutorial, youtube ansible tutorial, ansible module generate tutorial, learning ansible, What is ansible, Why do we use ansible, How does ansible Works, ansible basic tutorial pdf

Tags Ansible 9 Ansible Videos 7 Devops 45 Devops Videos 26 Trending 61

Next

Ansible Inventory Introduction - Ansible
 < Beginner Tutorials
 (https://www.learnitguide.net/2017/10/ansible-inventory-introduction-tutorial.html)

Previous

Install Openstack Horizon Dashboard -
 Openstack Tutorial >
 (https://www.learnitguide.net/2017/09/install-openstack-horizon-dashboard.html)

POST A COMMENT

Disqus Facebook Blogger

0 Comments learnitguide

1 Login ▾

Recommend Tweet f Share

Sort by Newest ▾



Start the discussion...

LOG IN WITH

OR SIGN UP WITH DISQUS ?

Name

Be the first to comment.

Subscribe Add Disqus to your site Add Disqus Add Disqus' Privacy Policy Privacy Policy Privacy Policy