Journal of Advanced Research in Intelligence Systems and Robotics
Volume 2, Issue 2 - 2020, Pg. No. 5-8
Peer Reviewed Journal

Article

# Ansible Framework for Various Platforms

Kapil Kokcha[1], Alisha Goyal[2], Ankita Joshi[3]

[1]B.Tech Scholar, [2,3]Assistant Professor, Department of CSE/IT Global Institute of Technology, Jaipur, India.

## I N F O

**Corresponding Author:**
Kapil Kokcha, Department of CSE/IT Global Institute of Technology, Jaipur, Indi.
**E-mail Id:**
16egjcs088@gitjaipur.com
**How to cite this article:**
Kokcha K, Goyal A, Joshi A Ansible Framework for Various Platforms. *J Adv Res Intel Sys Robot* 2020; 2(2): 5-8.

Date of Submission: 2020-04-03
Date of Acceptance: 2020-05-10

## A B S T R A C T

With the growing technology in the Industrial period of 4.0 we also have to grow and work fast. With or without the presence of human efforts we have to set up various labs and platforms without any delay. The high demand of cloud computing have increased the no of end users exponentially and those users have to be managed with or without an organization. The same is the case with web-applications and mobile applications development and deployment where daily the taste of end user may change and we have to serve the user with keeping in mind about all the demands and possibility in the short time period. The idea is to setup all the required components and to build a comfortable working environment for manual and automation work. [1] We can either setup each and every thing individually or we can work towards a single goal i.e to fulfil the users need and requirements. All this can be achieved by majorly two ways, either we work on various platforms to perform similar objective or we work on objective and our goal and let Ansible do the rest of the work. All this with the secure and protected environment of ansible.

**Keywords:** Ansible, Industry 4.0, Application development, Security

## Introduction

With the growing technology in the industrial era of Technology 4.0, we have to be quick and efficient with the setting up the labs to application development and deployment. Ansible is the best tool for our objective. The best part of ansible is that its open-source automation tool and platform. Ansible can be used for various tasks such as application development and positioning, conformation supervision, innumerable type of services for provisioning and communication network. Time, Accuracy and Consistency are important these days. Real time operations are the part of day to day life and we can't achieve all these together in real time by manual operations. Automation is the need of time and ansible is the best tool to achieve and deploy it.

Automation simplifies and makes easy the complex tasks, by not just making developers' jobs more manageable and efficient but permitting them to focus and concentrate on other tasks that add value to an organization. In additional confrontations, it releases up time and intensifications efficiency and accuracy. Ansible is one of the speedily rising tool in the field of automation.

We have to set up our processing unit in order to maintain the communication amid device to device, machine to machine, device to machine, or machine/device to cloud and for this the obligation of agile and reliable way of decreasing the software development life cycle as well as multi-platform development of software is must. The reduction in time, space and efforts between the application improvement and operation in real life scenario can be obtained using DevOps. The main aim of DevOps is to improve collaboration, integration, communication for the real world problems. Ansible is easy to learn and deploy as the ansible framework is written in Python and the playbooks are scripted using (YAML) YAML Ain't Markup Language.

## Configuration Management

In Ansible Configuration Management is the term used in order to preserve product configuration and keeping the record about software and hardware with complete detailed description of genuine enterprise product.

This type of information contains the exact versions and updates of the installed software packages and address of hardware devices with location to which these updates are made. For e.g. If we want to connect the latest type of HTTPD/WebService on all obtainable machines of our enterprise, then it will be a complex task to go at each machine manually and apply the updates to each machine in less time.

With the help of ansible we can just install all the required services whether be HTTPD/WebService or any other service in one go on all the available machines. The ansible playbooks and inventory helps us to achieve our goals and the best part is they are simple to write.

All we have to do is list out the IP addresses or the Host name of our nodes in the inventory and write an ansible playbook to install Servers and run the ansible playbook from your control machine & it will be installed on all host machines.

## Methodology

**Working:** Its mechanism is by joining the nodes using the SSH secure shell and pushing out small programs also called ansible modules. Now these pushed programs are executed and after the completion of the execution these files are uninvolved. The inventory is used in order to maintain all the nodes. And playbooks are designed in a way that it works on particular targeted inventory groups. And that's how the hosts are configured.

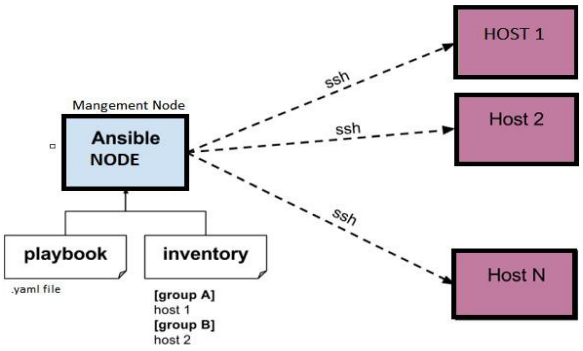


**Figure 1.Configuration**

**Playbook:** Playbook are the files in which code for the targeted machines are written. Theses playbooks are written in YAML format. YAML is the abbreviation for YAML Ain't Markup Language. Playbooks are one of the important characteristics of ansible and tells the ansible how to work and what to work. They are like a to-do list for Ansible that contains a list of tasks.[2]

Playbooks holds all the steps that a user wants to run on a particular host machine. Playbooks are run sequentially and they are the brains of Ansible and works on all the required cases. Each playbook can contain one or more than one plays in it. The plays are the set of statements that tells what to do and how to do on which machine. Playbooks are written in YAML language and it is strictly typed language. By strictly typed we mean that it follows syntax similar to python and we have to pay an extra care and attention while writing the code in it. Editor is not a barrier while writing a YAML file, we can choose any type of editor from notepad to sublime-text. Just open the editor, follow the syntax and start writing the code. The example of a playbook is give below.

```
---
- hosts: all
  vars_files:
        - i_touch_ansible.yml
  vars_prompt:
     - name: x
       prompt: "enter the vg name to be given"
       private: no


  tasks:
     - lvg:
         vg: "{{ u }}"
         pvs: "{{ r }}","{{ s }}"
         pesize: "{{ t }}"
```

**Figure 2.Playbook**

**Connection:** The nodes are connected with each other using SSH secure shell. The communication is totally secure and encrypted with algorithms like RSA(Rivest, Shamir, Adleman). It is a cryptographic network protocol used to secure network connections over an indiscreet network. To use SSH, we must use an SSH client to connect to the SSH server.[3] SSH have the capability to perform all task that be done by file transfer protocol (FTP). The communication is completely secure by SSH as it encrypts the data before sending and on receiver the data is decrypted. User do not have to worry about all these steps as SSH will automatically perform all these tasks in the back-end itself and user can carry their usual work. That's how the complete safe communication is carried out by ansible without the user being aware about any of these process.

**Provisioning:** The endowment is the process of setting up the environment and infrastructure. It helps to collect and connect all phases of the software. It can mechanize, manage and connect all parts of sdlc. The collection of resources and set up of environment is the initial step in the application development. No matter on which platform we are working, it may be the bare metal machine or the wired or wireless servers all have to go through provisioning in order to get automated. Ansible is capable of providing almost every platform, from cloud platforms to virtualization, from bare-metal servers to network dependent systems.

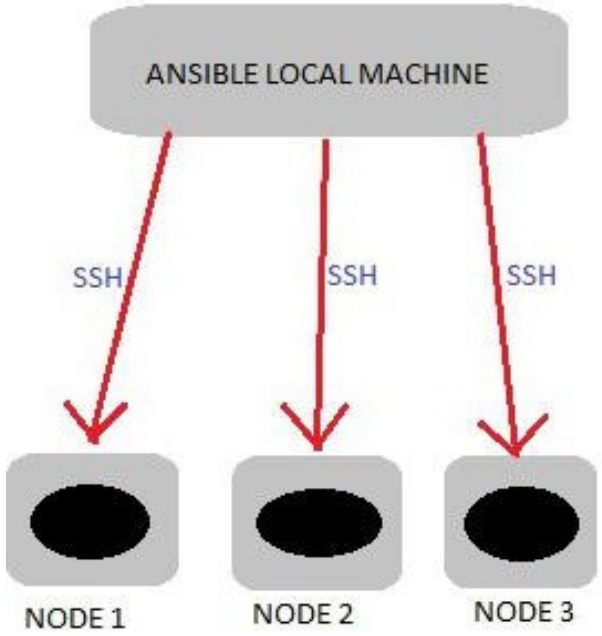


**Figure 3.SSH-Communication**

It also provides the parallel processing and tools can be connected using pipelines.

Ansible makes the establishment method flawless and easy. We can easily provision the configuration of a system, development of an application, programming and automation languages, and can make it easy as of human understandable form.

The physical server is always present behind the virtual and cloud platforms and sometimes the correct resources for these platforms is important and they can also be provided with ansible provisioning.

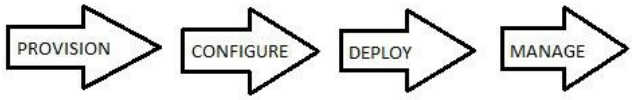


**Figure 4.Provisioning**

**The tip of the Ansible Iceberg:** Ansible is countless with the prebuilt playbooks and the tasks with in them, but ansible can be used dynamically also and this mode is called ad-hoc command mode. With the help of ad-hoc mode we can use and run the same commands on different devices. To increase the power we have different plugins and filters. Ansible is great tool for network related tasks. The best part is, it is much easier to configure and playbooks are also easy to write and configure. All these are easier to understand, quickly to grasp and learn, much powerful then other similar tools.

## Conclusion

Automation plays a major role from setting up a lab to software development with agile methodology. Every updation should be fast as possible with the property to be edited as soon as any bugs gets reported. With Ansible playbooks these setups can be done with fewer code. Its architecture is simple and effective. It works by connecting to your nodes and pushing small programs to them. These programs make the system comply with a desired state, and, when they have finished their tasks, they are deleted. Ansible can help us with the use placement, task automation, and formation supervision.

It can also help us in running the tasks in particular sequence where ever required and all this can be done on different machine all together.

An example is if we have lots of web servers or machines for our work. Ansible perform the updation those to all those servers and machines one at a time with the removal of older versions and disabling them in the system. So in short we can handle complex tasks with a tool which is easy to use and apply.

Unlike other similar tools like Puppet or Chef it doesn't necessitates any middle agent. Instead of that ansible have the secure shell called SSH and it is used and we need to install this shell on the system which we want to connect and use. Ansible is written in most famous language of recent times called python and ansible playbooks are written in yaml but it follows mostly all syntax of python which makes it easy for user to understand, learn and implement. We do not require any special setup like a client server environment setup, we just need that python should be installed on the main machine.

## Future Scope

With virtualization on the rise along with cloud computing breakthroughs, application development, data science the standard on-premise services, and infrastructure management has become highly scalable and simple. The need for managing a huge number of servers within or outside the organization has become the most important work in recent times.

When various servers, machines and their applications running on different platforms are to be accomplished and maintained, then this will become complex and time consuming and ansible automation will provide the help of time saving and efficiency.

This can range from supervision installations that support small to large applications and maintaining the desired state of a huge number of servers.

Compared to other similar tools like Puppet or Chef, Ansible tool is very fast because of its agent-less deployment and communication. Which means it does not use an agent on the remote host. Instead, Ansible uses SSH secure shell which is assumed to be present on all the systems you want to manage.

Ansible is built on Python language, in contrast to the Ruby as base foundation of Puppet and Chef. The learning curve for Ansible is not as deep and hard as Chef/Puppet because of excellent official documentation. Playbooks are pretty easy and straightforward and have clear structure which makes it easy to learn, implement and understand. Ansible provides better security by using SSH/SSH2 secure shells which are far better in several ways than the security provided by Chef and Puppet.

Ansible is most quickly rising tool in automation, configuration management, provisioning, application development etc. It have the capability to take over the famous tools of its category like Chef and Puppet. The reason it got so much admiration is that it is easy to learn, implement, simplicity of code, agent-less nature and various other factors. The ansible community is still improving and working on it. It can run on various platforms and in term of operating system, it supports Windows and Mac-OS but it is still in development phase for these operating systems and regularly modules are added for them but it is fully functional for Linux kernel based systems and automation tools like ansible is the future for almost every platform.

## References

1. Ansible : Ansible I, 2020. Ansible Documentation. [online] Docs.ansible.com Available at:<https://docs.ansible.com>
2. Ansible Project : Docs.ansible.com. 2020. Ansible Documentation - Ansible Documentation. [online] Available-at: <https://docs.ansible.com/ansible/latest/index.html>.
3. Network: Searchnetworking.techtarget.com.2020. Networking Information, News and Tips - Searchnetworking. [online] Available at: <https://searchnetworking.techtarget.com/>.