**Ansible:** is a configuration management and often placed in the vicinity as of CHEF, PUPPET, and SALT. Configuration management means configuring the destination and making all the things ready for the deployment. Ansible can also be used for deployment.  
Whenever multiple servers are involved, then the process of prepping the services in a specific order is termed as orchestration.

Configuration Management tools work on two methodologies:

* **Pull Based:** Chef and puppet use pull base, in pull based agents are installed on the destination machine and the agent poll the central service for any changes. Once changes are detected the agent pulls down and implements the changes on the agent machines.  
  Pull Based mechanism works as below:
  + Changes are made in the configuration management scripts.
  + Changes are pushed to the central service.
  + Agent on the Destination server checks for changes periodically.
  + Agent connects to central service.
  + Pulls down the changes.
  + Agent executes the changes on the destination server.
* **Push Bashed:** In push based no agent installation is required on the destination server. Advantage of push based is that we control when the changes are to be pushed to the server. The push based mechanism works as follows
  + Changes are made in the configuration Management scripts.
  + Scripts are run.
  + Ansible makes ssh connection and executes the steps in the scripts(aka Playbooks)
* **Installation of Ansible:** Ansible can be installed by various ways, one of the easiest one is to use the default package manager to install.  
  For centos/redhat: use yum package manager as it resolves the required dependencies automatically.  
  yum –y install Ansible
  + **Ansible install its configuration files under :** /etc/Ansible
    - **Ansible.**cfg: the default configuration files for Ansible and has all the information on where to do the logging and which inventory file to pick. I would recommend going through this file once to get the idea for which configurations can be customized.
    - Whenever Ansible runs it looks up the configuration file in following places in the below order:
      * File specified in ANSIBLE\_CONFIG environment variable
      * Current directory (./Ansible.cfg)
      * Users home directory (~/.ansible.cfg)
      * /etc/Ansible/Ansible.cfg

Hosts file: this is the file that contains the information for the hosts that can be configured using Ansible. Which host file Ansible will use is defined in Ansible.cfg under parameter hostfile=<location\_to\_file>  
This can be over written in ansible command line by passing ansible -I <file\_name>

* Once the Ansible is installed, hosts can be defined in the host file defined as below
  + [group\_name]  
    <server-alias> ansible\_host=<server-ip> ansible\_connection=ssh remote\_user=<username-to-remote> ansible\_ssh\_pass=<password>
  + If the remote user is common for all the machine it can be defined in the ansible.cfg file and same user will be used for connection.
  + Check the connection to host running ping module  
    ansible –m <module\_name> ping <host\_name> or <host\_group>
  + We can run command module on the destination to run linux commands  
    ansible –m command –a “tail /var/<log\_file>
  + Below is sample ansible playbook, create a file playbook.yml and paste in below contents  
    -name: Configure Web Servers with NGINX  
     hosts: <defined in the host-file>  
    sudo: true  
    tasks:  
     -name: install nxginx  
     -apt: name=nginx update\_cache=yes  
     -name: copy custom html files  
     -copy: src=<location\_of custom\_html\_file> dest=/usr/share/nginx/html/index2.html mode=0644  
     -name=restart nginx  
     -service: name=nginx state=restarted
  + To run the playbook, execute ansible-playbook <playbook\_name>
  + Note: since the ansible uses the cfg file present in the current directory first, the way I like to do it is for every play I put in a ansible.cfg that contains a custom log location for every playbook, just to make things clean, we will be discussing ansible.cfg in detail later on.