*DevOps*

**DevOps**

(Tutorial – Channel Name: Logiclabstech recordings on Google Drive)

**First Published On: 24th Jun 2021**

**Last Updated On: 24th Jun 2021**

**DevOps**

|  |
| --- |
| **CHEF** |

|  |  |
| --- | --- |
| Configuration Management Tool    **Ansible**      **Installation and configuration** (<https://www.youtube.com/watch?v=kE-6KDyf-0o&list=PLBGx66SQNZ8aPsFDwb79JrS2KQBTIZo10&index=32>)     1. Create 3 EC2 Amazon Linux 2. Download the package   *sudo su*  *wget* [*https://dl.fedoraproject.org/pub/epel/epel-release-latest-7.noarch.rpm*](https://dl.fedoraproject.org/pub/epel/epel-release-latest-7.noarch.rpm) (download epel package)  *ls*     1. Install the package   *yum install epel-release-latest-7.noarch.rpm*  *yum update -y*   1. Install the components   *yum install git python python-level python-pip openssl ansible –y*  *ansible –y*   1. Configuring hosts file   Create group and add private ips of Node Instances  *vi /etc/ansible/hosts*     1. Configuring cfg file   In order to reflect the changes done in hosts file, we uncomment one entry so that the Server understands to consider the Inventory IPs assigned in hosts  *vi /etc/ansible/ansible.cfg*     1. Add user so that it does the task of root user (do this on all the 3 servers, add same user and password in all instances)   *adduser <<any name>>*  *passwd <<name given in above line>>*  (enter password)   1. Switch user (on all 3 servers)   *su - <<provided username>>*   1. Give root access to the new user   Switch back to root user  *visudo*    Scroll down  add another entry as shown by typing *i* and save after adding user using :wq    Do this on all the 3 servers   1. Install httpd   *sudo yum install httpd –y*     1. Change another Configuration file   vi /etc/ssh/sshd\_config    Scroll down and do below changes      Do this for all 3 servers   1. Restart all 3 servers      1. Connect your master with nodes   Make sure that all the nodes are logged in with new created user  *ssh <<private IP of node instance>>*     1. Now, for each remote access from CLI, it is asking for password, which we want to remove the prompt and build trust relationship   Make sure you are connected using same user in all the 3 instances  *ssh-keygen*  Hit enter wherever it prompts    *ls*     1. Copy id\_rsa.pub in remaining 2 servers   ssh-copy-id <<user>>@<<privateip>>    Now you can connect to any node without prompting for password   1. Common commands   *ansible all --list-hosts* (to get all the nodes connected)    *ansible <<groupname>> --list-hosts* (to get all the nodes in the specified group)    Three types of running commands on Nodes   1. Ad-hoc 2. Modules 3. Playbooks   **Ad-hoc Commands**    *ansible <<group name>> -a “<<argument>>”* (runs ls command on all the nodes of the group)  *ansible demo -a “ls”*    *ansible <<groupname>>[<<index>>] -a “<<argument>>”*  *ansible demo[0] –a “touch filez”*  *ansible demo –a “sudo yum install httpd -y”*    *Ansible demo –ba “yum remove httpd -y”*    **Modules**    *ansible demo –b –m yum –a “pkg=httpd state=present”*      *ansible demo –b –m service –a “name=httpd state=started”*    *Ansible demo –b –m user –a “name=xyz”*    *cat /etc/passwd*    *ansible demo[-1] -b -m copy -a “src=xyz dest=/temp”*    **Setup**  *ansible demo –m setup*    Gives you all the details of the nodes | **Configuration Management Tool**  Configuration Management – System Admin who is responsible for managing the servers.  Eg. Installing OS, installing softwares (MS Office, Sophos, Browsers, etc), Configurations (configuring outlook, AD policies, permissions)  If there is a demand of installing an UBUNTU OS, admins will install the server and configure the IP, **but there is a problem**, if the demand is for 1000 servers, it will be very difficult to install everywhere one by one. Here configuration management tool comes into picture  There are two mechanisms of configuration management   1. Push Based:  * Here, a code is written in a file (Infrastructure as a Code) and is pushed across all the server, the code will have instructions of all the installation and configuration. This will do the configurations across all the 1000 servers in 10 mins. * You use this if you need complete control and centralized   Eg: Ansible, Saltstack.   1. Pull Based:  * Here, the instances check by connecting to the server if there are any new versions for the software, and if difference is found, the software is updated. * Here, advantage is if new instances are added, it will automatically sync the configurations and update itself.   Eg: Chef, Puppet  Ansible   * It is a configuration management tool * Acquired by REDHAT in 2015 * Ansible – it was use in ancient times, A device that is used to communicate between stars or earth and star without any delay * Built on Python * Scripts written in YAML is called Playbook   Advantages   * Can be used OnPrem and on Cloud * Uses YAML, easy to write * Agentless   **Three types of running commands on Nodes**   1. **Ad-hoc**   Running commands directly  This has a disadvantage – there is no idempotency i.e. each time same command is given it will execute, it is not smart enough to understand that the job was done in 1st command itself.   1. **Module**   Single Work   1. **Playbook**   Multiple modules  **Ad-hoc Commands**  So, if there are 1000 nodes, this command will create filez file on all servers  **-b acts same as sudo**  demo - group name  -b - sudo privileges  -m - module  yum - module name  pkg=httpd - package name  state=present - install  Now if you run this same command second time, it will give a message that already installed. This is called idempotency  To start the service  To create a user to all nodes of specified group  Verify on the node instance that user is created  The user entry should be at the end.  To copy the specified file to last node[-1] of the group specified  Setup   * Setup is a module present in the nodes * It is responsible to store all the state (installed files, configuration, etc). * This is the reason it maintains idempotency i.e. it knows that the files are already present in the node and no need to reinstall them |