|  |
| --- |
|  |
|  | **Ansible Server & Host Installation/configuration:**  **Part1: Install and configure Ansible Server** |
|  | 1. Build an EC2 Machine with SSH Port open e.g. Amzon EC2 system. |
|  | 2. Create a user e.g. ansible admin (ansadmin) with sudo access, password authentication enabled. This user will have privelege to SSH into remote hosts using its public key. |
|  | 3. Login as root ( sudo su -) |
|  | 4. Craete user: useradd ansadmin |
|  | 5. Create password: passwd ansadmin |
|  | 6. add user to use sudo command by editing sudoers file: vi /etc/sudoers or visudo (late is recommended): add command " ansadmin ALL=(ALL) NOPASSWD: ALL " |
|  | { |
|  | ## Same thing without a password |
|  | # %wheel ALL=(ALL) NOPASSWD: ALL |
|  | ansadmin ALL=(ALL) NOPASSWD: ALL |
|  | } |
|  |  |
|  | 7. Enable Password authentication: vi /etc/ssh/sshd\_config (comment out "PasswordAuthentication yes" and comment in "PasswordAuthentication no") |
|  | { |
|  | # To disable tunneled clear text passwords, change to no here! |
|  | PasswordAuthentication yes |
|  | #PermitEmptyPasswords no |
|  | #PasswordAuthentication no |
|  | } |
|  | 8. Reload the service sshd by restarting the sshd daemon so that it does not exit on changing user or system restart. |
|  | RHEL/CENTOS - sudo systemctl restart sshd.service (Ubuntu/Debian - systemctl restart ssh.service) & service sshd reload |
|  | { |
|  | # systemctl restart sshd.service |
|  | [root@ip-172-31-94-236 ~]# service sshd reload |
|  | Redirecting to /bin/systemctl reload sshd.service |
|  | } |
|  | 9. Check the password authentication by connecting the Server without user name and keypair in mobaxterm by just entering ip into remote host and click ok, which will prompt: |
|  | login as: ansadmin |
|  | ansadmin@54.165.210.110's password: |
|  | 10. Install ansible: yum install ansible and it might redirect to link for ansible installation |
|  | 11. Login as ansadmin and verify pwd is /home/ansadmin |
|  | 12. Generate SSH Keypair: ssh-keygen to generate private & public key. We will copy public key into hosts so that ansadmin from ansible server can manage host systems. |
|  | 13. Also verify the content of /etc/ansible/ which should be " ansible.cfg hosts roles " |
|  |  |
|  | **Part2: Install Hosts** |
|  | 1. Prepare a host system e.g. RHEL EC2 System |
|  | 2. Login as root user & Create a user (User & Password should be same as craeted in Ansible Server) |
|  | 3. Add user to sudoers: visudo and shift+G will lead to the end of the file where we can add "ansadmin ALL=(ALL) NOPASSWD: ALL" |
|  | (Note: In RHEL system an entry for ec2-user is observed at the end of file but not seen in Amazon-linux system probably ec2-user is built-in.) |
|  | 4. Enable Passworh auth: vi /etc/ssh/sshd\_config, sudo systemctl restart ssh.service & service sshd reload |
|  |  |
|  | **Part3:**  Copy Public Key of user ansadmin from ansible server to user ansadmin in hosts. |
|  | 1. Login to Ansible Server as ansadmin: cd .ssh/ and copy the public key in the remote host using below command: |
|  | "ssh-copy-id ansadmin@<private-ip=address of remote host system>" which will ask for confirmation as well as verify the success by "ssh ansadmin@<ip of the host system>" |
|  | 2. Success to ssh menas ansadmin user can manage remote host envionment using Ansible Playbooks. |
|  |  |
|  | **Part4:**  Add system to /etc/ansible/hosts file, test ping connection and run the playbooks. |
|  | 1. test the connectivity between Ansible Server and hosts: "anisble all -m ping" |
|  | 2. Create a playbook e.g. install.yml to install the software packages |
|  | 3. Verify the playbook: " ansible-playbook install.yml --check" or "ansible-playbook install.yml -C" |
|  | 4. Run the playbook: "ansible-playbook install.yml" |
|  | 5. Login the host system and verify the installation of the software packages.  **Part5:**  Manage the Ansible Server - This is done by using it as localhost system.  1. Add entery in the hosts file either as "localhost" or private ip address of the Ansible Server.  2. Copy public key to local host: [ansadmin@ansible-server .ssh]$ ssh-copy-id localhost  3. Test connection: [ansadmin@ansible-server .ssh]$ ssh localhost |
|  |  |
|  | **Key Note:**  Ubuntu host and difference between " Service sshd restart and reload" |
|  | 1. Ubuntu system does not craete a home directory by default for a user e.g. in Centos/RHEL family ansadmin use creation creates /home/ansadmin but not in Debian family. |
|  | { e.g. we create a user ansadmin and run root@ip-172-31-84-168:~# su - ansadmin |
|  | su: warning: cannot change directory to /home/ansadmin: No such file or directory |
|  | $ pwd |
|  | /root (No home directory for anssdmin |
|  | $ |
|  | } |
|  | So we will have to create a home directory for ansdmin |
|  | root@ip-172-31-84-168:~# cd /home/ |
|  | root@ip-172-31-84-168:/home# ls -al |
|  | drwxr-xr-x 2 root root 4096 Aug 19 09:03 ansadmin |
|  | drwxr-xr-x 4 ubuntu ubuntu 4096 Aug 19 08:58 ubuntu |
|  | chown -R ansadmin:ansadmin /home/ansadmin/ |
|  | drwxr-xr-x 2 ansadmin ansadmin 4096 Aug 19 09:03 ansadmin |
|  | drwxr-xr-x 4 ubuntu ubuntu 4096 Aug 19 08:58 ubuntu |
|  | root@ip-172-31-84-168:/home#su - ansadmin |
|  | $ pwd |
|  | /home/ansadmin (Home directory created for user ansadmin) |
|  | $exit and |
|  | # User privilege specification |
|  | root ALL=(ALL:ALL) ALL |
|  | ansadmin ALL=(ALL:ALL) NOPASSWD: ALL |
|  | & |
|  | root@ip-172-31-84-168:/home# vi /etc/ssh/sshd\_config |
|  | # To disable tunneled clear text passwords, change to no here! |
|  | PasswordAuthentication yes |
|  | service sshd reload |
|  | Copy public key and confirm the connecyion by ssh into the host. |
|  |  |
|  | Generally speaking, restart will terminate the service in question and restart it; reload will only reload the configuration file. |
|  | Since reload only reloads the config, it will cause less disruption to existing activities and currently open connections; users might not even notice that it was run. |
|  | However, depending on what server we're talking about, some options may not be changeable using reload; and if the server is using too much memory, etc, |
|  | it may be necessary to use restart to force it to start from a clean slate. |

**Ansible Installation**

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

**Prerequisites:**

1. An AWS EC2 instance for Ansible Server
2. A client or host

**Installation steps:**

1. Install Ansible Add a EPEL (Extra Packages for Enterprise Linux)third party repository to get packages for Ansible

$rpm -Uvh<https://dl.fedoraproject.org/pub/epel/epel-release-latest-7.noarch.rpm>

$yum install ansible -y

# ansible --version

1. Create a new user for ansible administration & grant admin access to user (Master and Slave)

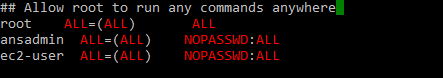
$ useraddansadmin

$ passwdansiadmin(password) and passphrase- ( e.g. test@123)

To delete: userdel

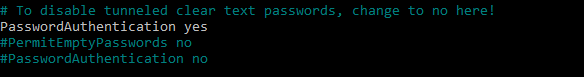
1. Add usersansadmin to sudoers file.

# nano /etc/sudoers



1. Enable password authentication by editing /etc/ssh/sshd\_config so that Ansible user can run playbook without password authentication

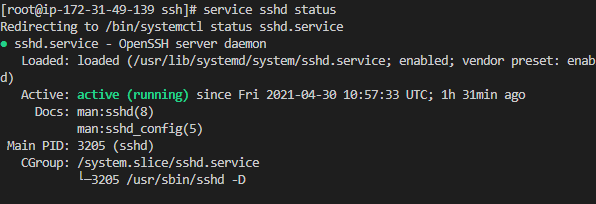
# nano /etc/ssh/sshd\_config



# servicesshd start



# servicesshd status



$sshd –t (to check what is the issue in case sshd do not run)

1. Login as aansadmin user on master and generate ssh key (Master) which is copied to the host.

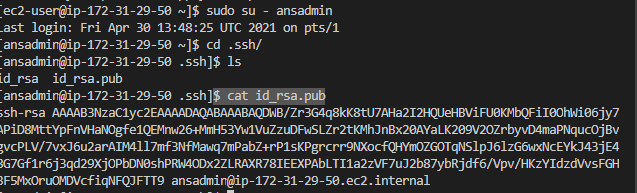
Login to EC2 Instance and then run as ansadmin

ec2-user@ip-172-31-29-50 ~]$ sudosu–ansadmin

Generate SS Key:

[ansadmin@ip-172-31-29-50 ~]$ aws-keygen

$ cat id\_rsa.pub

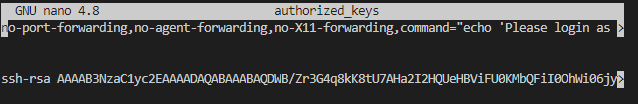


Login to Cluster:

[ec2-user@ip-172-31-86-242 ~]$ ssh -i ~/.ssh/id\_rsa[ubuntu@api.cluster.demo.com](mailto:ubuntu@api.cluster.demo.com)

$ sudosu–

root@ip-172-20-46-73:~/.ssh# nanoauthorized\_keys



Now test password less authentication to access host from ansible server

([ansadmin@ip-172-31-29-50 ~]$ssh -i id\_rsaroot@Host IP Address)

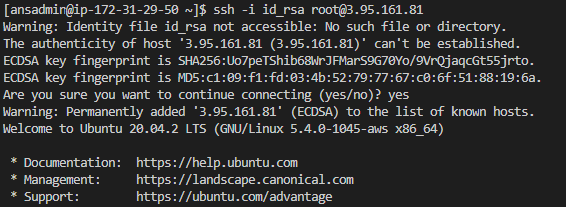
Host IP address will be private in case of same VPC and Public in case different VPC. Since the cluster is in different VPC so will use Master Node Public IP Address.

[ansadmin@ip-172-31-29-50 ~]$ ssh -i id\_rsa[root@3.95.161.81](mailto:root@3.95.161.81)

root@ip-172-20-46-73:~# hostname

ip-172-20-46-73

( It was successful login without password as ip-172-20-46-73 is Cluster Master Node Private IP Address)



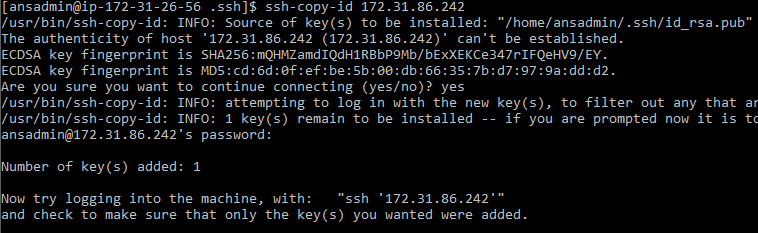
1. Copy Pub keys into the Authorized Host of all ansible client nodes :

e.gssh-copy-id <host private ip>

In case Host is in different VPC then we will have have to copy and paste after login to that host.

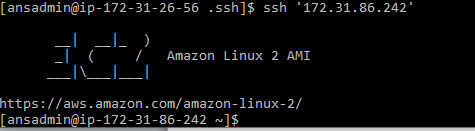
e.g. Kubernetes Master as a host which will be accessed by ansible server through ssh without password

$ ssh-copy-id 172.31.86.242



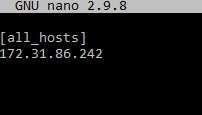
1. Verify we able to login to client or host system without password

$ ssh '172.31.86.242'

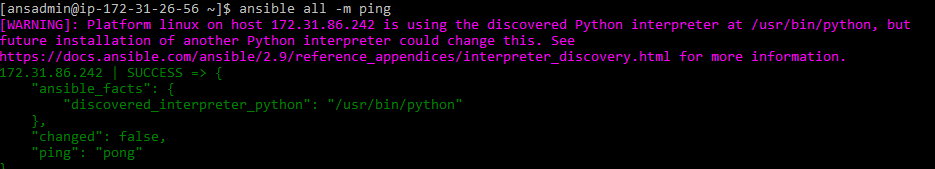


1. Create a host inventory in the Ansible Server file at /etc/ansible/hosts and check ansible commands

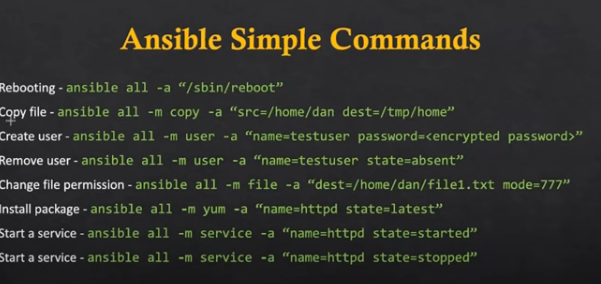
$ sudonano /etc/ansible/hosts



$ ansible all -m ping



Sample commands:



Create a hello.html file in the Ansible server and copy to the host.

“ansible all -m copy -a "src=/ dest=/"

Verify the destination e.g. home of the ansible client.

Login to ansible client and check the ansible home path

[root@ip-172-31-86-242 ~]# cd /home/ansadmin/

[root@ip-172-31-86-242 ansadmin]# pwd

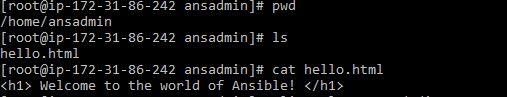
/home/ansadmin

$ ansible all -m copy -a "src=/home/ansadmin/hello.html dest=/home/ansadmin"

Its copied to host home as status shows Success with “Changed=true”



Login to Host to verify:



If we run the command again the it will show success but “Changed = false”

