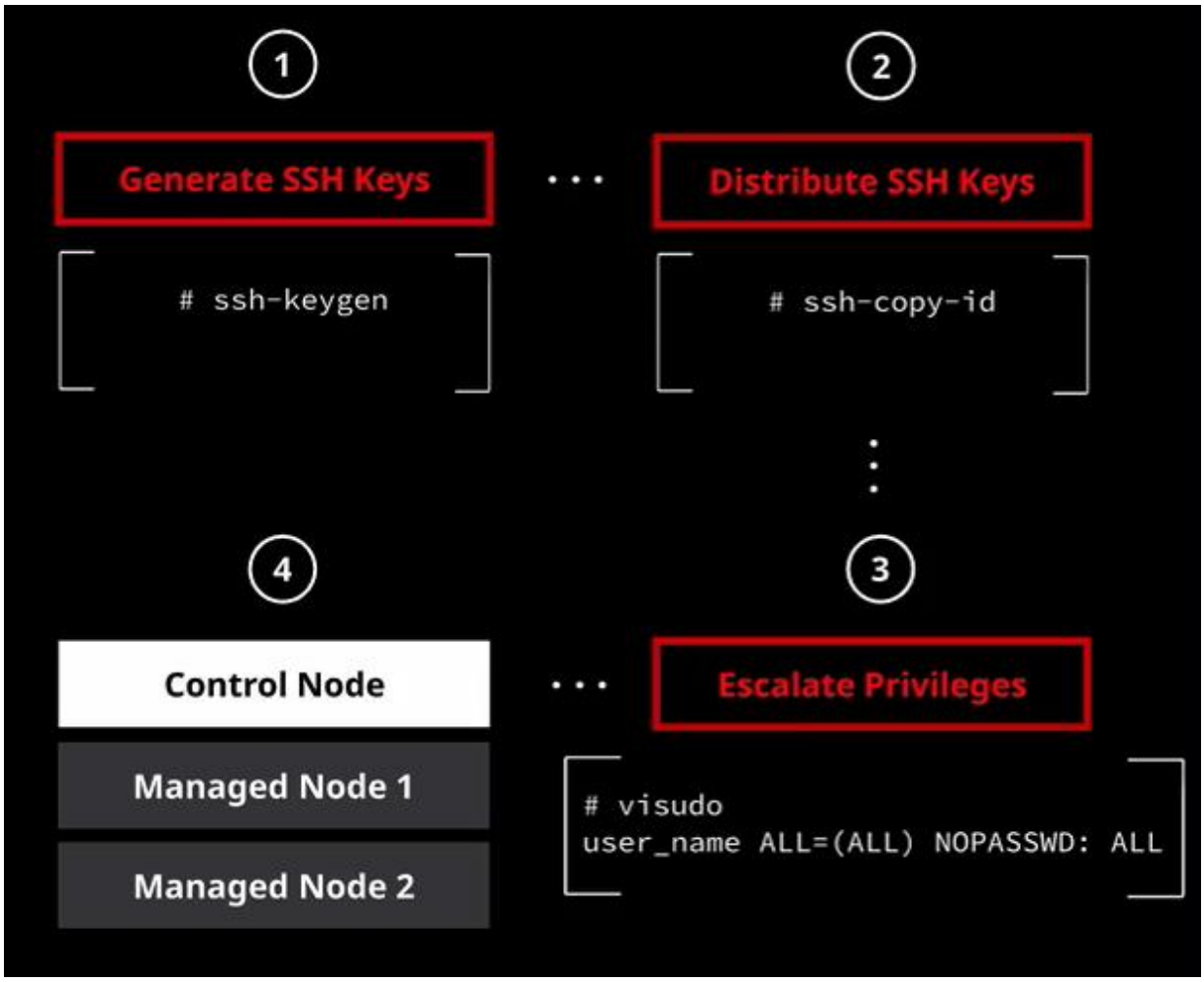
Configure Ansible Manage Node

Create user with ansadmin in all node and set password
Add ansadmin user in sudoers file in all node and master too

$ ssh-keygen
$ ls -l
$ cat /home/ansadmin/.ssh/id_rsa.pub
$ ssh-copy-id ansadmin@server1
$ ssh-copy-id ansadmin@server2
Note: Copy public ken in all node
$ ssh ansadmin@server1
Note : test with all node



Host Inventory File

What is Inventories?

- An *inventory* defines a collection of hosts that Ansible will manage.
- The Ansible inventory file defines the list of hosts, upon which Ansible works.
- Host inventories can be defined in two different ways.
    - Static: - A *static* host inventory can be defined by a text file.
    - Dynamic: - A *dynamic* host inventory can be generated by a script or other program as needed, using external information providers.

Inventories

- Defaults location of hosts file: /etc/ansible/hosts
- The default location of the hosts file can be set in /etc/ansible/ansible.cfg
- It can be specified using -i option when running ansible.
- The file can contain individual hosts, groups of host, groups of groups and host and group level variables.
- It can contains variable that determine how you connect to a host.
- Inventories files may contains hosts, patterns, groups and variable.
- Multiple inventory files may be specified
- A static inventory file is a text file that specifies the managed hosts that Ansible targets.
- You can write this file using a number of different formats, including INI-style or YAML.
- Host groups allow you to more effectively run Ansible against a collection of systems. In this case, each section starts with a host group name enclosed in square brackets ([]). This is followed by the host name or an IP address for each managed host in the group, each on a single line.

Static Inventory Examples

192.168.1.1
db1.jetking.com

[db-servers]
db1.jetking.com
db2.jetking.com
db2.jetking.com

[db-servers]
db[1:3].jetking.com

[web-servers]
192.168.1.10
192.168.1.11
192.168.1.12

[web-servers]
192.168.1.[10:12]

[test-servers]
Servera
Serverb
Serverc

[prod-server]
Server[a:c]

[prod:children]
web-servers
db-servers

Verifying Inventory

- $ ansible web-servers - -list-hosts
- $ ansible web-servers –m ping

Manually Define- Location of the Inventory

- With the **--inventory *PATHNAME*** or **-i *PATHNAME*** option, where **PATHNAME** is the path to the desired inventory file.
- $ ansible –i inventory db-servers –m ping
- $ ansible –i inventory db-servers:active –m ping

ANSIBLE CONFIGURATION FILES

- The behaviour of an Ansible installation can be customized by modifying settings in the Ansible configuration file.
- Ansible chooses its configuration file from one of several possible locations on the control node.

Precedence Ansible Configuration files

- Possible location of Ansible configuration files (in order processed)
    - ANSIBLE_CONFIG (environment variable)
    - ansible.cgf (in the home directory)
    - ~ / .ansible .cfg (in the home directory)
    - /etc /ansible /ansible.cfg (Default location)

Common Ansible Configuration:

- The ansible-config command can be used to view configuration:
- list -Prints all configuration option
- dump -Dumps configuration
- view- View the configuration file

Setting in Configuration File:

- The Ansible configuration file consists of several sections, with each section containing settings defined as key-value pairs.
- Section titles are enclosed in square brackets.
- For basic operation use the following two sections:
    - **[defaults] :** sets defaults for Ansible operation
        - [defaults]
        - inventory = ./inventory
        - remote_user = user
        - ask_pass = false

- **[privilege_escalation] :** configures how Ansible performs privilege escalation on manage hosts
  - [privilege_escalation]
  - become = true
  - become_method = sudo
  - become_user = root
  - become_ask_pass = false

Connection Setting:

- By default, Ansible connects to managed hosts using the SSH protocol. The most important parameters that control how Ansible connects to the managed hosts are set in the **[defaults]** section.
- By default, Ansible attempts to connect to the managed host using the same user name as the local user running the Ansible commands. To specify a different remote user, set the **remote_user** parameter to that user name.
- If the local user running Ansible has private SSH keys configured that allow them to authenticate as the remote user on the managed hosts, Ansible automatically logs in.
- If that is not the case, you can configure Ansible to prompt the local user for the password used by the remote user by setting the directive **ask_pass = true**.

Escalating Privileges

- For security and auditing reasons, Ansible might need to connect to remote hosts as an unprivileged user before escalating privileges to get administrative access as root.
- This can be set up in the **[privilege_escalation]** section of the Ansible configuration file.
- To enable privilege escalation by default, set the directive **become = true** in the configuration file.
- The **become_method** directive specifies how to escalate privileges.
- Several options are available, but the default is to use **sudo**. Likewise, the **become_user** directive specifies which user to escalate to, but the default is root.

- If the **become_method** mechanism chosen requires the user to enter a password to escalate privileges, you can set the **become_ask_pass = true** directive in the configuration file.

Configuration File Comments:

- There are two comment characters allowed by Ansible configuration files: the hash or number sign
- (#) and the semicolon (;).
- The number sign at the start of a line comments out the entire line. It must not be on the same line
- with a directive.
- The semicolon character comments out everything to the right of it on the line. It can be on the
- same line as a directive, as long as that directive is to its left.

  Custom Configuration File:
  interpreter_python = auto
  inventory = /home/ansadmin/ansible/inventory/inv.ini
  roles = /etc/ansible/roles

Note: if you are not setting all parameter, remaining will take from default location
Also interpreter in default location- /etc/ansible/ansible.cfg