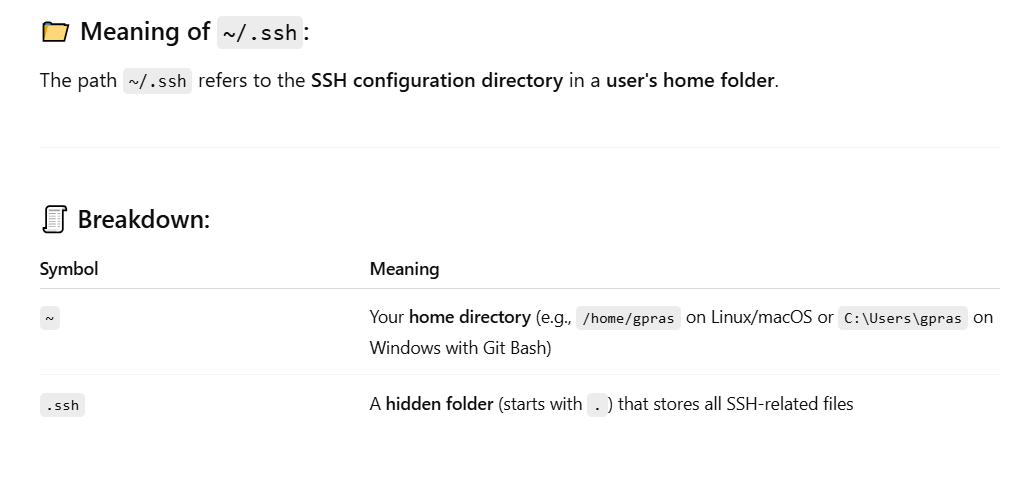
Let’s see How to connect EC2 instances from our Local Computer using SSH KEY(private or public) instead Access Token



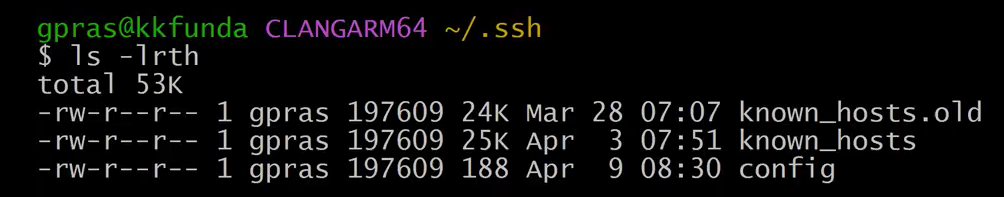


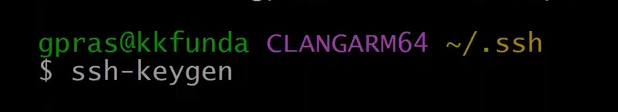
The terminal prompt in the above screenshot shows that the current user is:

* gpras (username)
* on the machine kkfunda
* using the shell in the directory ~/.ssh (which is the SSH configuration directory in the user's home directory)

This means you're currently located inside the .ssh folder, which typically contains SSH keys and configuration files such as:

* id\_rsa (private key)
* id\_rsa.pub (public key)
* known\_hosts
* config



Next execute the command ssh-keygen .   


This command is used to **generate a new SSH key pair** (a public and private key) by default for secure authentication.

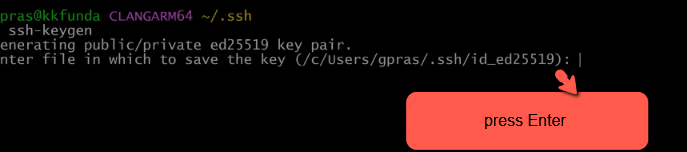
🔐 Why it's used:

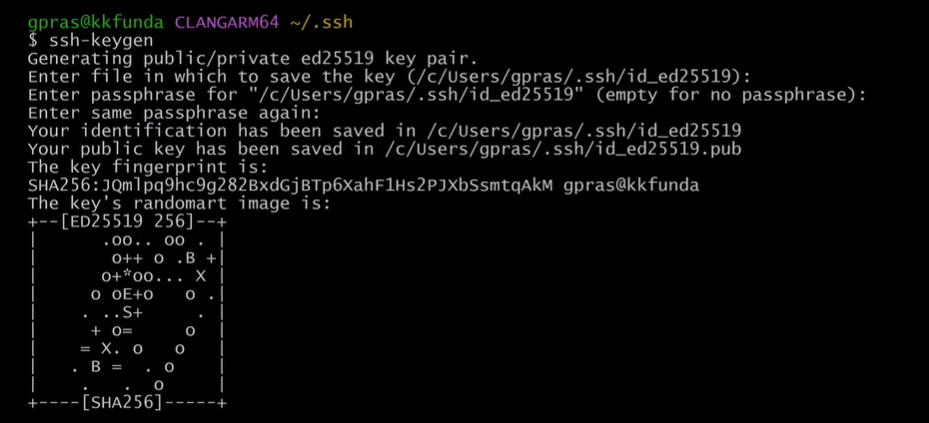
* To securely connect to remote servers without typing a password every time.
* Commonly used for Git, remote SSH login, cloud access (e.g., AWS), and automation scripts.

🧾 What it does:

When you run ssh-keygen, it:

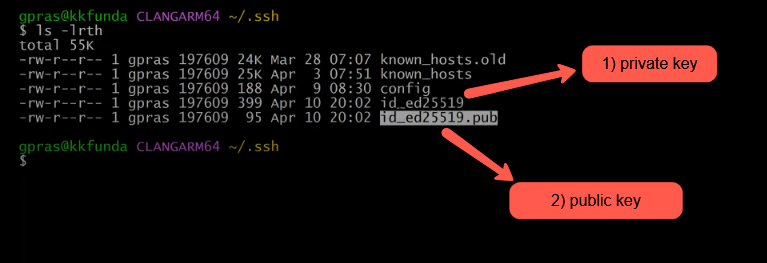
1. Asks for a file name to save the key (defaults to ~/.ssh/id\_rsa).
2. Optionally prompts you to set a passphrase (adds extra security).
3. Creates two files:
   * id\_rsa → your private key (keep safe and never share).
   * id\_rsa.pub → your public key (can be shared with servers you want to access).





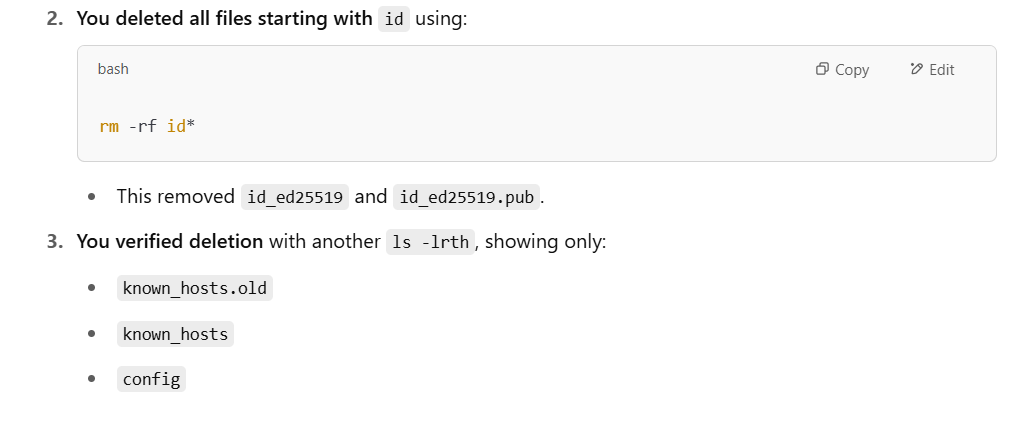
Next clear the screen  

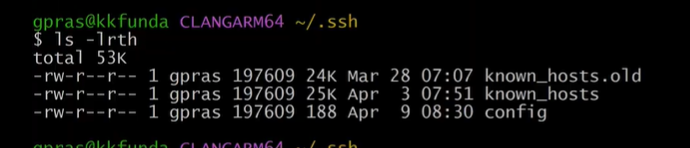

Next Execute the command ls –lrth and you will see two files 1) Private Key file and 2) public key file



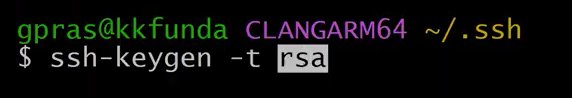
We’ll delete the existing private and public keys, create a new SSH key pair, and use it to connect to the EC2 server.

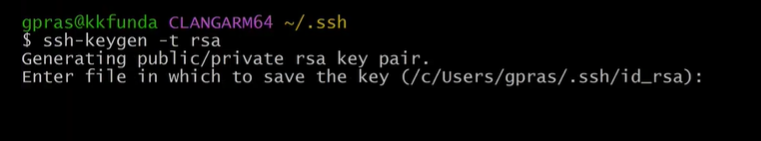
Execute the command rm –rf id\* to delete the existing private and public keys as shown in the below image  
  



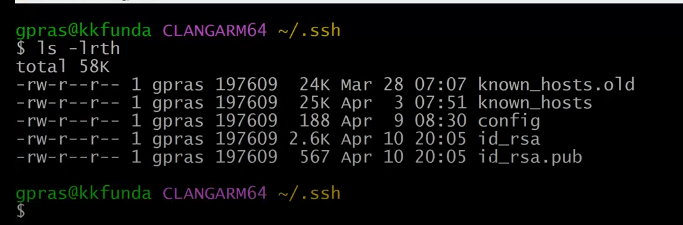
Lets generate the private and pubic keys using rsa algorithm so lets execute the below command



Press Enter It will generate private and public keys using rsa algorithm  


Execute the command clear and ls –lrth





As you can see in the above image private and public keys using rsa algorithm got generated

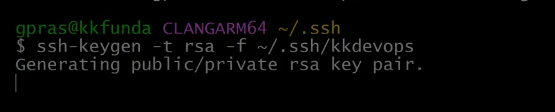
* id\_rsa → Private key (keep it safe and never share it)
* id\_rsa.pub → Public key (you can upload this to your EC2 server)

We can generate SSH keys(private & public keys) using a custom name instead of the default RSA filename.

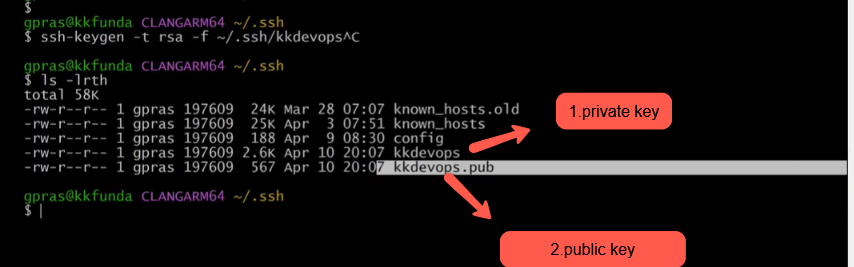
1. Execute the command rm –rf id\*
2. Then execute the command ssh-keygen -t rsa -f ~/.ssh/kkdevops

Shown in the below image

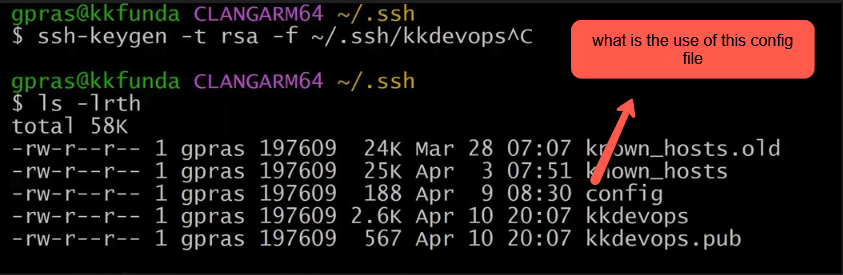


Keep Pressing Enter  


As shown in the below image private key and public key is successfully got created with custom name kkdevops



This is How we can generate the ssh-keygen

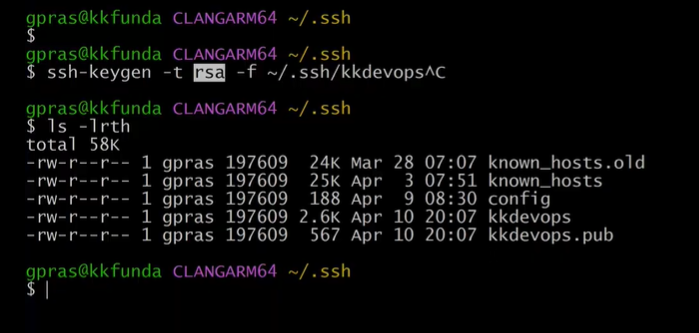


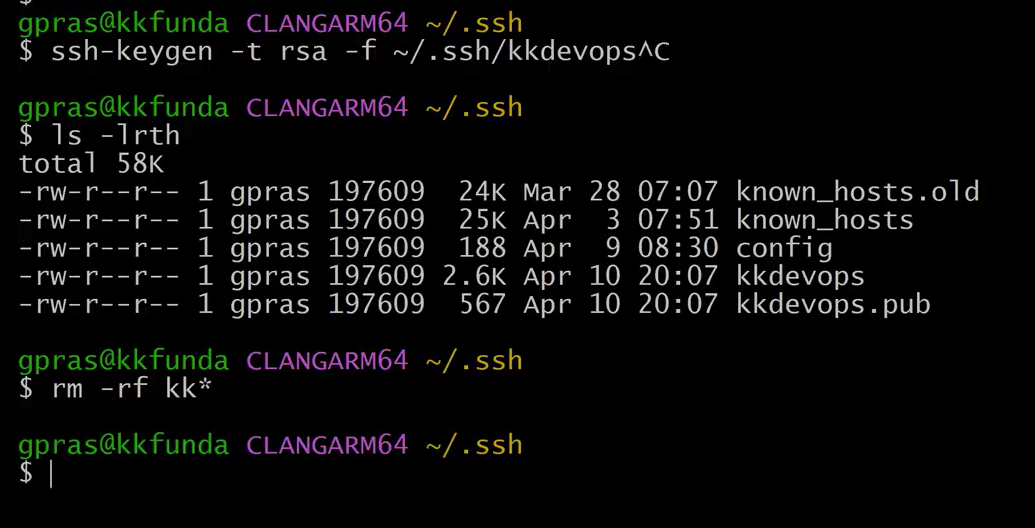
📄 Purpose of the config file in the .ssh folder:

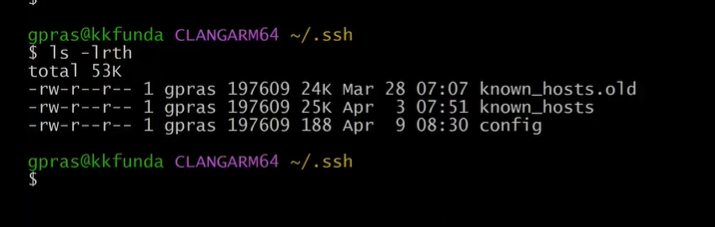
The ~/.ssh/config file is used to simplify and customize SSH connections.

1. Assign different SSH keys for different hosts
2. Use shortcuts for long SSH commands
3. Specify custom ports, users, and options per host

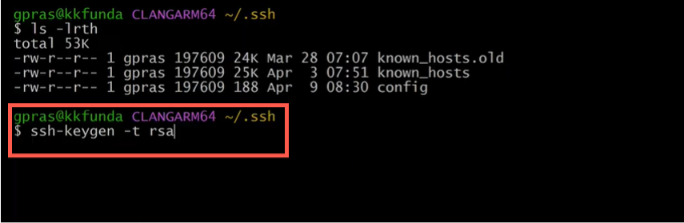
Check the video from minutes 26::00::00 discussed about config file and known host file

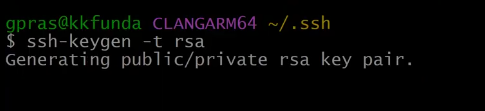


Lets remove kkdevops alias name of rsa algorithm by executinf the command rm -rf kk\* shown in the below image   


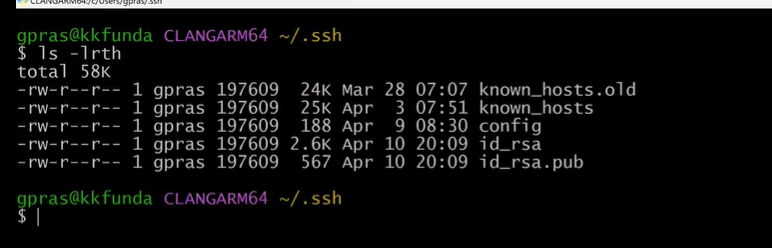
Clear the screen  


Lets again generate the rsa alogorithm as shown in the below image



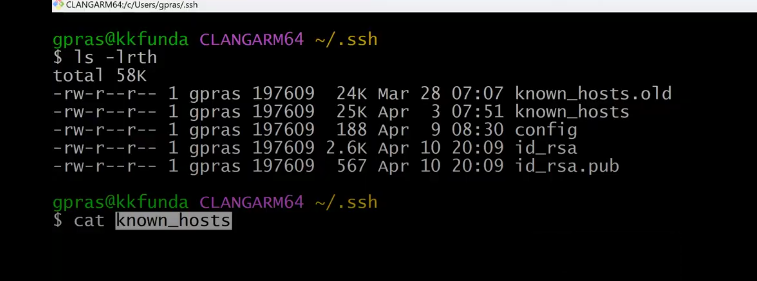


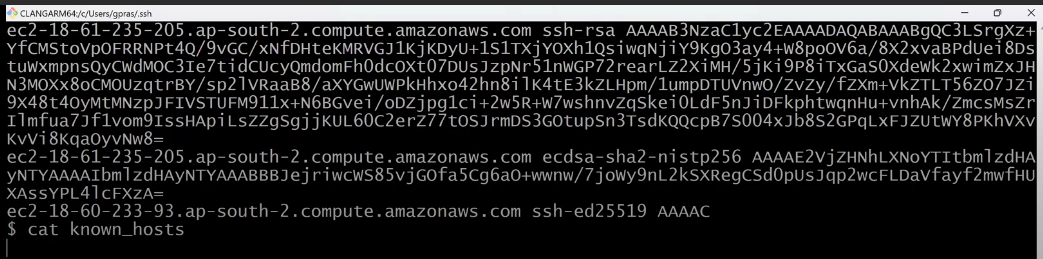
Now we have public key and private key as shown in the below image



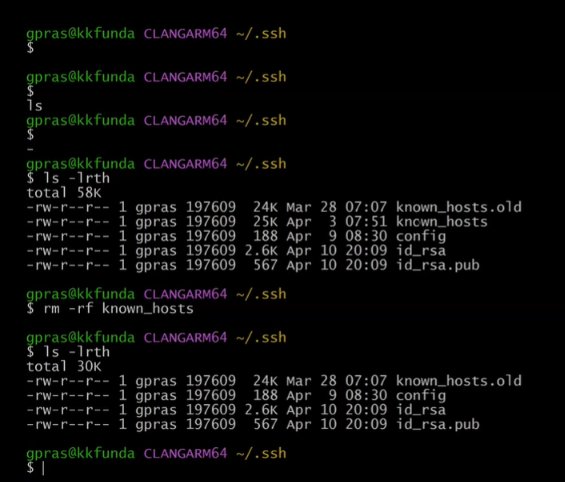
Check the video from 26::00::00 to 48:00:00 discussing about the purpose of config and known\_hosts files

Known hosts file stores the host information

<https://www.dropbox.com/login>



Remove the known host and generate the ssh again



Git Interview Question (videos starts at 50:00:00)