----------**Generating GPG key**----------

1. cat /etc/lsb-release
   Output: DISTRIB_ID=Ubuntu
   DISTRIB_RELEASE=16.04
   DISTRIB_CODENAME=xenial
   DISTRIB_DESCRIPTION="Ubuntu 16.04 LTS"

   ----------Installation----------
2. sudo apt-get update
3. sudo apt-get install gnupg-agent
4. sudo apt-get install git
   ----------Generate Key----------
5. gpg –gen-key
   Output: gpg (GnuPG) 1.4.20; Copyright (C) 2015 Free Software Foundation, Inc.
   This is free software: you are free to change and redistribute it.
   There is NO WARRANTY, to the extent permitted by law.
   Please select what kind of key you want:
   (1) RSA and RSA (default)
   (2) DSA and Elgamal
   (3) DSA (sign only)
   (4) RSA (sign only)
   Your selection? **4** ==> **Select 4 i.e., RSA(sign only)**
   RSA keys may be between 1024 and 4096 bits long.
   What keysize do you want? (2048) **4096** ==>**Type 4096**
   Requested keysize is 4096 bits
   Please specify how long the key should be valid.
   0 = key does not expire
   <n> = key expires in n days
   <n>w = key expires in n weeks
   <n>m = key expires in n months
   <n>y = key expires in n years
   Key is valid for? (0) **0** ==> **Select 0**
   Key does not expire at all
   Is this correct? (y/N) **y** ==> **yes**

   You need a user ID to identify your key; the software constructs the user ID from the Real Name, Comment and Email Address in this form:
   "Heinrich Heine (Der Dichter) <heinrichh@duesseldorf.de>"

   Real name: **Gajendra** ===> **Name to define your key**
   Email address: **gajendra_ramesh@thbs.com** ===> **email address**
   Comment:
   You selected this USER-ID:
   "Gajendra <gajendra_ramesh@thbs.com>"

   Change (N)ame, (C)omment, (E)mail or (O)kay/(Q)uit? **O** ==> **Select okay**
   You need a Passphrase to protect your secret
   Enter Passphrase:
   Retype Passphrase:

   The Process of generating the key begins..., in order to generate a key the system should be busy. To keep the system busy,

open another terminal:

1. **Install Stress:** sudo apt-get install stress
2. **Run this command:** stress -c 2 -i 1 -m 1 --vm-bytes 550M -t 60s

Once generation is completed, you will see the confirmation displaying pub id, key fingerprint and uid.

----------Display Keys----------

6. gpg –list-keys

   Output: /home/user/.gnupg/pubring.gpg
   ---
   pub 4096R/EB2CCF15 2016-11-17
   uid Gajendra <gajendra_ramesh@thbs.com>

7. gpg -a --output gajendra.private.asc --export-secret-keys

8. ls -ltr

   Output: total 52
   -rw-r--r-- 1 user user 8980 Nov 16 14:19 examples.desktop
   drwxr-xr-x 2 user user 4096 Nov 16 15:32 Desktop
   drwxr-xr-x 2 user user 4096 Nov 16 15:32 Videos
   drwxr-xr-x 2 user user 4096 Nov 16 15:32 Templates
   drwxr-xr-x 2 user user 4096 Nov 16 15:32 Public
   drwxr-xr-x 2 user user 4096 Nov 16 15:32 Pictures
   drwxr-xr-x 2 user user 4096 Nov 16 15:32 Music
   drwxr-xr-x 2 user user 4096 Nov 16 15:32 Documents
   drwxr-xr-x 2 user user 4096 Nov 16 17:19 Downloads
   drwxrwxr-x 2 user user 4096 Nov 17 11:35 gpg_backup
   -rw-rw-r-- 1 user user 3451 Nov 17 14:17 Gajendra.private.asc

----------Generating Secret Keys----------

9. gpg -c --cipher-algo AES256 Gajendra.private.asc
10. gpg --list-secret-keys

    Output: /home/user/.gnupg/secring.gpg
    ---
    sec 4096R/EB2CCF15 2016-11-17
    uid Gajendra <gajendra_ramesh@thbs.com>

11. ls -ltr

    Output: total 56
    -rw-r--r-- 1 user user 8980 Nov 16 14:19 examples.desktop
    drwxr-xr-x 2 user user 4096 Nov 16 15:32 Desktop
    drwxr-xr-x 2 user user 4096 Nov 16 15:32 Videos
    drwxr-xr-x 2 user user 4096 Nov 16 15:32 Templates
    drwxr-xr-x 2 user user 4096 Nov 16 15:32 Public
    drwxr-xr-x 2 user user 4096 Nov 16 15:32 Pictures
    drwxr-xr-x 2 user user 4096 Nov 16 15:32 Music
    drwxr-xr-x 2 user user 4096 Nov 16 15:32 Documents
    drwxr-xr-x 2 user user 4096 Nov 16 17:19 Downloads
    drwxrwxr-x 2 user user 4096 Nov 17 11:35 gpg_backup
    -rw-rw-r-- 1 user user 3451 Nov 17 14:17 Gajendra.private.asc
    -rw-rw-r-- 1 user user 2730 Nov 17 14:20 Gajendra.private.asc.gpg

12. gpg –list-keys

    Output: /home/user/.gnupg/pubring.gpg
    ---
    pub 4096R/EB2CCF15 2016-11-17

uid Gajendra <gajendra_ramesh@thbs.com>
sub 4096R/DF2CEJ19 2016-11-17

----------Public key----------

13. gpg --export --armor gajendra_ramesh@thbs.com > Gajendra-pubkey.asc
14. ls -ltr

Output: total 60
-rw-r--r-- 1 user user 8980 Nov 16 14:19 examples.desktop
drwxr-xr-x 2 user user 4096 Nov 16 15:32 Desktop
drwxr-xr-x 2 user user 4096 Nov 16 15:32 Videos
drwxr-xr-x 2 user user 4096 Nov 16 15:32 Templates
drwxr-xr-x 2 user user 4096 Nov 16 15:32 Public
drwxr-xr-x 2 user user 4096 Nov 16 15:32 Pictures
drwxr-xr-x 2 user user 4096 Nov 16 15:32 Music
drwxr-xr-x 2 user user 4096 Nov 16 15:32 Documents
drwxr-xr-x 2 user user 4096 Nov 16 17:19 Downloads
drwxrwxr-x 2 user user 4096 Nov 17 11:35 gpg_backup
-rw-rw-r-- 1 user user 3451 Nov 17 14:17 Gajendra.private.asc
-rw-rw-r-- 1 user user 2730 Nov 17 14:20 Gajendra.private.asc.gpg
-rw-rw-r-- 1 user user 1637 Nov 17 14:27 Gajendra-pubkey.asc

15. mkdir gpg_backup/
16. cp --preserve Gajendra* gpg_backup
17. cd gpg_backup/
18. ls -ltr
19. Output: total 12
-rw-rw-r-- 1 user user 3451 Nov 17 14:17 Gajendra.private.asc
-rw-rw-r-- 1 user user 2730 Nov 17 14:20 Gajendra.private.asc.gpg
-rw-rw-r-- 1 user user 1637 Nov 17 14:27 Gajendra-pubkey.asc
20. cd
21. tar cvpf backup_.gnupg.tar .gnupg
22. ls -ltr
23. mv backup_.gnupg.tar gpg_backup/
24. cd gpg_backup/
25. ls -ltr
26. cd ..
27. mkdir keys
28. mv Gajendra* keys/
29. ls -ltr

The Keys are stored in the key directory, Share the public key (Not Private Key) with the only person who has to decrypt your file.

----------**XXX**----------

----------**Encrypting & Decrypting File**----------

In order to send a encrypted file to a specified recipient, the recipient's public key should be imported to the keyring.

----------importing keyring----------

1. cd
2. gpg --import manjesh.public.asc

Output: gpg: key 80D5357B: public key "Manjesh P (MyAccount) <manjesh_raj@thbs.com>" imported

gpg: Total number processed: 1
gpg: imported: 1 (RSA: 1)

3. gpg –list-keys ==> **the keys are visible if imported successfully**
4. gpg –edit-key 80D5357B
5. **gpg>** trust
   select 5
6. **gpg>** sign in

   ----------encrypting a file----------
7. gpg -e -r manjesh_raj@thbs.com testfile
   file is encrypted
8. gpg -d testfile.gpg

----------XXX----------