

## Assignment-2

### 1. Generating GPG key pair and sending the publickey as a file to the recipient

```
Activities Terminal
guru@ubuntu: ~/Desktop/receiver$ gpg --gen-key
gpg (GnuPG) 2.2.19: Copyright (C) 2019 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.

Note: Use "gpg --full-generate-key" for a full featured key generation dialog.

GnuPG needs to construct a user ID to identify your key.

Real name: guru
Name must be at least 5 characters long
Real name: gurusaran
Email address: gurusarandasar1998@gmail.com
You selected this USER-ID:
" gurusaran <gurusarandasar1998@gmail.com>"

Change (N)ame, (E)mail, or (O)key/(Q)uit? o
We need to generate a lot of random bytes. It is a good idea to perform
some other action (type on the keyboard, move the mouse, utilize the
disks) during the prime generation; this gives the random number
generator a better chance to gain enough entropy.
We need to generate a lot of random bytes. It is a good idea to perform
some other action (type on the keyboard, move the mouse, utilize the
disks) during the prime generation; this gives the random number
generator a better chance to gain enough entropy.
gpg: key C34B4F8CF2FBE75 marked as ultimately trusted
gpgs: directory /home/guru/.gnupg/openpgp-revocs.d/ created
gpgs: revocation certificate stored as /home/guru/.gnupg/openpgp-revocs.d/05064913D4A722F6973D6C8FC34B4F8CF2FBE75.rev'
public and secret key created and signed.

pub rsa3072 2020-09-22 [SC] [expires: 2022-09-22]
    05064913D4A722F6973D6C8FC34B4F8CF2FBE75
uid
    gurusaran <gurusarandasar1998@gmail.com>
sub rsa3072 2020-09-22 [E] [expires: 2022-09-22]

guru@ubuntu: ~/Desktop/receiver$ gpg --armor --output gurupubkey.gpg --export gurusarandasar1998@gmail.com
guru@ubuntu: ~/Desktop/receiver$ ls
gurupubkey.gpg
guru@ubuntu: ~/Desktop/receiver$ cat gurupubkey.gpg
-----BEGIN PGP PUBLIC KEY BLOCK-----

mQGNBFP9pWBOADNLNKGXGA9gdzeFv2VLY/rGc/T/Cx00ZEDspqrkTAFEQ0rsah
QbKdb/g59f8xJXBBdVd8pKwZ2P+6Ys6DQ08UzWq0z0N5hb7F45K7J5zAKLXrZ
/kNecA3sLHudlo8L1d04f79UMRCbWfPaSCDcp0p501IhcJaEQNm0YNB5ssof
xW3WVU+XWYQBP7CDyHBAQp5SCHWtY5x1e+08E/YH+ZLmbg/V8LBUK50Ny5EM
wS5V1YSE71NAAKVovzNaqotnrbDEZLSb4EXUpqBPRVfjMhdyad6zpd7dVMAO
Cq2kEuyj21k/Ltambbaat1A0AK10fz2Wlrfgeu5J+3s1nfddcl32qCfL
TspokbBAPhXncnezfH+P0HPC42EFmb0JH1XcLGHuLo772d4q2KebH290A0DnZ
A4Fhr553DnpJka39n43Vferj5LAQ/WL26uEmwCF85n1NncvN/LIzP5x+gokNlp
qTfV9V9gncdC4lQ0AABQp3Uy6dNncwP1Dxdx31CzfyYwskYmHcnxv01K4
QdGtVtSLmwbT63AdQpEwK4HATQp8sk1Tkt19pc8BjPDS+8Bv+oDQC2hNA
0w1bWuJAB3nAAULC0gHAGVYcgK1CIEfGIDAQIAQIkgAAKCRD0S0+8Bv+dxZ+
DADGpBMG07506zKdrvwux+raQa3c1jZG0o1rXoHBL734ta0wBMBWVNY30D
gd4boC1LHm1P8SpK2H824fD1Xc0yVc5Kx13N+T9dCPWPFfEuH0Bae2KE
oHLXKSeP8XnCaShgTPCgctPfor80n75ubPFOCjrtEXLUZu3G0Ww4j3DL+I
2BVNP9+6K2r+Xddqu2pk0Tg33PrfQ5APFULD4C9v+YtebPPDPON1zfkQrId7d1
1PAR9Vl0IECuz/K39lQvgv2Hj+3o4KetjCSMKExozHr6L8Y70UL44h4nq+8W
4W9SLyy083+CS3PqQZCF+ku0byaf6a2eH0pgeKx1bW2HfLsePlay7PmbW
1av1+5qne3Q1D63mWLG4LSQ2EL4980SP+4IfI29JF4R1bCvN9sV0R8Dacag
0FV8MB5MACU1Qha7qAs0tW+H1p4W7F49enW1RusTMAAGds1GrTYqu1H0UHQ3
Bok5AY8E2NA0WEPANSPYocJTeLVUA7X0CCFTUjQhGduf/qK3Z0NPFLdnyNnx
HtGr15fRy3t4epTVCC3Xhu3j8d0X09fEBM0obbi/c4eeu85GlepxA9VZot
BC0yQqFABQQUQndH131Lf5ytToXdsd0gXWxEIT7h81j7nte/cbrhXRYW64KX
39n/b0U3yWq46Pase084Unusj1ac1LV3sJUS0X9GZ7F33g164jP6PsreCL3ip
vV09pRc4qns1scCmfantQ/Yb9y48IguJmWKSPEBPC31Fr1CLTNNZvclvlg
0R0V5TnJ/Sa1TfDme/qFrm99F1VWm9ahuaMSj0Kv4K1vLT0D3EP31ge12LS
aQhjetQ/rKlk3ny1dZNMK3Tvngh5jnz54s1oAonjLhyeZL2Z08GpX1rheyZfn
h13RA/rZuk80h8nxbYld2LWyg8N03pXjLEJl1M1d0e2hLTnq0zCRBo7z4Kz
214kVbJdL1/MPpWbAqA81Q08H83gmfLEBQZtE951nXWpWj0EPVPL/
vndFAl9pWNCowFCQPC2AAcGk0wETPvPL/vnxRtQv4Au1y3YVNUZ1Nj3d1MSf
F70p8EBxAcNKR0GVKxEkK89K3FATUJ34kccmouuQ7814E4Htw5DPEHBedb
1VU85ndP/pvaCm04/DCBpVfYbnjWfK9d81NtX1Q8C/OhThNTBPCD0u5uPTT
j34Fe2p4qQPFneayP23apN6Cv0B8ZufZfpg2L35VnIELAL0ECXAP67
t6ox7/UGszKIL15NT1P9wD0E1M475h15NEXFKKCNJHgE9KX+52y3xLNCCKkK
N58x8YBa53JgWk0K0jVWRL74uNBGrY1DD9a0kbyuXogeEzIMURku18FT5YdXx
NLt1T5cSRLt63Hm1WgK/9vFLP0G607XALCPaeQJmWbLbWpDQMH4u+X8B05L
U108pVtWkKxaj1ALG0B3fG0B8EFYaaLVm9HwJ0d3TTHB3xVpdx5x/njW
+R4/Beubhcdg5ck/D31uTcd5MTUUAU12AhWyrJ56Lz1N
=14Pe
-----END PGP PUBLIC KEY BLOCK-----

guru@ubuntu: ~/Desktop/receiver$
```

```
Activities Terminal
guru@ubuntu: ~/Desktop/receiver$ ls
gurupubkey.gpg
guru@ubuntu: ~/Desktop/receiver$ cat gurupubkey.gpg
-----BEGIN PGP PUBLIC KEY BLOCK-----

mQGNBFP9pWBOADNLNKGXGA9gdzeFv2VLY/rGc/T/Cx00ZEDspqrkTAFEQ0rsah
QbKdb/g59f8xJXBBdVd8pKwZ2P+6Ys6DQ08UzWq0z0N5hb7F45K7J5zAKLXrZ
/kNecA3sLHudlo8L1d04f79UMRCbWfPaSCDcp0p501IhcJaEQNm0YNB5ssof
xW3WVU+XWYQBP7CDyHBAQp5SCHWtY5x1e+08E/YH+ZLmbg/V8LBUK50Ny5EM
wS5V1YSE71NAAKVovzNaqotnrbDEZLSb4EXUpqBPRVfjMhdyad6zpd7dVMAO
Cq2kEuyj21k/Ltambbaat1A0AK10fz2Wlrfgeu5J+3s1nfddcl32qCfL
TspokbBAPhXncnezfH+P0HPC42EFmb0JH1XcLGHuLo772d4q2KebH290A0DnZ
A4Fhr553DnpJka39n43Vferj5LAQ/WL26uEmwCF85n1NncvN/LIzP5x+gokNlp
qTfV9V9gncdC4lQ0AABQp3Uy6dNncwP1Dxdx31CzfyYwskYmHcnxv01K4
QdGtVtSLmwbT63AdQpEwK4HATQp8sk1Tkt19pc8BjPDS+8Bv+oDQC2hNA
0w1bWuJAB3nAAULC0gHAGVYcgK1CIEfGIDAQIAQIkgAAKCRD0S0+8Bv+dxZ+
DADGpBMG07506zKdrvwux+raQa3c1jZG0o1rXoHBL734ta0wBMBWVNY30D
gd4boC1LHm1P8SpK2H824fD1Xc0yVc5Kx13N+T9dCPWPFfEuH0Bae2KE
oHLXKSeP8XnCaShgTPCgctPfor80n75ubPFOCjrtEXLUZu3G0Ww4j3DL+I
2BVNP9+6K2r+Xddqu2pk0Tg33PrfQ5APFULD4C9v+YtebPPDPON1zfkQrId7d1
1PAR9Vl0IECuz/K39lQvgv2Hj+3o4KetjCSMKExozHr6L8Y70UL44h4nq+8W
4W9SLyy083+CS3PqQZCF+ku0byaf6a2eH0pgeKx1bW2HfLsePlay7PmbW
1av1+5qne3Q1D63mWLG4LSQ2EL4980SP+4IfI29JF4R1bCvN9sV0R8Dacag
0FV8MB5MACU1Qha7qAs0tW+H1p4W7F49enW1RusTMAAGds1GrTYqu1H0UHQ3
Bok5AY8E2NA0WEPANSPYocJTeLVUA7X0CCFTUjQhGduf/qK3Z0NPFLdnyNnx
HtGr15fRy3t4epTVCC3Xhu3j8d0X09fEBM0obbi/c4eeu85GlepxA9VZot
BC0yQqFABQQUQndH131Lf5ytToXdsd0gXWxEIT7h81j7nte/cbrhXRYW64KX
39n/b0U3yWq46Pase084Unusj1ac1LV3sJUS0X9GZ7F33g164jP6PsreCL3ip
vV09pRc4qns1scCmfantQ/Yb9y48IguJmWKSPEBPC31Fr1CLTNNZvclvlg
0R0V5TnJ/Sa1TfDme/qFrm99F1VWm9ahuaMSj0Kv4K1vLT0D3EP31ge12LS
aQhjetQ/rKlk3ny1dZNMK3Tvngh5jnz54s1oAonjLhyeZL2Z08GpX1rheyZfn
h13RA/rZuk80h8nxbYld2LWyg8N03pXjLEJl1M1d0e2hLTnq0zCRBo7z4Kz
214kVbJdL1/MPpWbAqA81Q08H83gmfLEBQZtE951nXWpWj0EPVPL/
vndFAl9pWNCowFCQPC2AAcGk0wETPvPL/vnxRtQv4Au1y3YVNUZ1Nj3d1MSf
F70p8EBxAcNKR0GVKxEkK89K3FATUJ34kccmouuQ7814E4Htw5DPEHBedb
1VU85ndP/pvaCm04/DCBpVfYbnjWfK9d81NtX1Q8C/OhThNTBPCD0u5uPTT
j34Fe2p4qQPFneayP23apN6Cv0B8ZufZfpg2L35VnIELAL0ECXAP67
t6ox7/UGszKIL15NT1P9wD0E1M475h15NEXFKKCNJHgE9KX+52y3xLNCCKkK
N58x8YBa53JgWk0K0jVWRL74uNBGrY1DD9a0kbyuXogeEzIMURku18FT5YdXx
NLt1T5cSRLt63Hm1WgK/9vFLP0G607XALCPaeQJmWbLbWpDQMH4u+X8B05L
U108pVtWkKxaj1ALG0B3fG0B8EFYaaLVm9HwJ0d3TTHB3xVpdx5x/njW
+R4/Beubhcdg5ck/D31uTcd5MTUUAU12AhWyrJ56Lz1N
=14Pe
-----END PGP PUBLIC KEY BLOCK-----

guru@ubuntu: ~/Desktop/receiver$
```

2. The public key is sent to Linux machine (sender) through mail
3. Using public key, message file is encrypted.

```
01:15 AM
gurusaran@guru:~/Desktop$ cd ~/Desktop/
gurusaran@guru:~/Desktop$ mkdir sender
gurusaran@guru:~/Desktop$ cd sender/
gurusaran@guru:~/Desktop/sender$ ls
gurupubkey.gpg
gurusaran@guru:~/Desktop/sender$ gedit msgtosend
bash: gedit: command not found
gurusaran@guru:~/Desktop/sender$ touch msgtosend
gurusaran@guru:~/Desktop/sender$ nano msgtosend
gurusaran@guru:~/Desktop/sender$ ls
gurupubkey.gpg      msgtosend
gurusaran@guru:~/Desktop/sender$ cat msgtosend
Hi, I am Guru Saran
gurusaran@guru:~/Desktop/sender$ gpg --import gurupubkey.gpg
gpg: /home/gurusaran/.gnupg/trustdb.gpg: trustdb created
gpg: key C1A84BCF7FFB75: public key "gurusaran <gurusarandasari1998@gmail.com>" imported
gpg: Total number processed: 1
gpg:    imported: 1
gurusaran@guru:~/Desktop/sender$ gpg --output msgencrypted --encrypt --recipient gurusarandasari1998@gmail.com msgtosend
gpg: 1EC5D708BF64E8: There is no assurance this key belongs to the named user
sub rsa3872/1EC5D708BF64E8 2028-09-22 gurusaran <gurusarandasari1998@gmail.com>
Primary key fingerprint: 8986 4013 D847 22FA 971D  8CAF C2AB AF3C F2F1 B87A
Subkey fingerprint: 1699 8408 584A 3142 818F  C1E9 1EC5 DF70 88F8 4018

It is NOT certain that the key belongs to the person named
in the user ID. If you really know what you are doing,
you may answer the next question with yes.

Use this key anyway? (y/N) y
gurusaran@guru:~/Desktop/sender$ ls
gurupubkey.gpg  msgencrypted  msgtosend
gurusaran@guru:~/Desktop/sender$ cat msgencrypted
-----BEGIN PGP MESSAGE-----
Version: 1.0
Comment: GnuPG v2.2.34
mQIwYQ==
=AD=
-----END PGP MESSAGE-----
-----BEGIN PGP SIGNATURE-----
Version: 1.0
Comment: GnuPG v2.2.34
mQIwYQ==
=AD=
-----END PGP SIGNATURE-----
```

4. Encrypted file is sent to receiver through mail
5. Using public key, the encrypted file is decrypted using public key.

[illegible]

**Signature verification:**

In sender, we created a public key for signature verification with mail.

Send the “msgtosendSigned” file to receiver

```
gurusaran@guru:~/Desktopender
File Actions
Current workspace: "Workspace 1"

guruPG needs to construct a user ID to identify your key.

Real name: venkat
Email address: gurusaran@gmail.com
You must choose the user ID to use:
"venkat.gurusaran@gmail.com"

Change (Name), (Email), or (O)kay/(Q)uit? o
We need to generate a lot of random bytes. It is a good idea to perform
some other action (type on the keyboard, move the mouse, utilize the
disks) during the prime generation; this gives the random number
generator a better chance to gain enough entropy.
We need to generate a lot of random bytes. It is a good idea to perform
some other action (type on the keyboard, move the mouse, utilize the
disks) during the prime generation; this gives the random number
generator a better chance to gain enough entropy.
gpg: key 3CD6F33E7EDAB09 marked as ultimately trusted
gpg: directory '/home/gurusaran/.gnupg/openpgp-revocs.d' created
gpg: revocation certificate stored as '/home/gurusaran/.gnupg/openpgp-revocs.d/59FF1DC268968ED
130CF2905CD6F33E7EDAB09.rev'
public and secret key created and signed.

pub  rsa3072 2020-09-24 [SC] [expires: 2022-09-24]
      59FF1DC268968ED130CF2905CD6F33E7EDAB09
uid          venkat.gurusaran@gmail.com
sub  rsa3072 2020-09-24 [E] [expires: 2022-09-24]

gurusaran@guru:~/Desktop/ender$ shasum -a 256 msgtossend | awk '{print $1}' > msgChecksum
gurusaran@guru:~/Desktop/ender$ gpg --output msgtossendSigned --sign msgChecksum
gurusaran@guru:~/Desktop/ender$ ls
guruspubkey.gpg msgChecksum msgencrypted msgtossend msgtossendSigned
gurusaran@guru:~/Desktop/ender$
```

At receiver side, verify the signature file.

The screenshot shows a terminal window with the following content:

```

guru@ubuntu:~/Desktop/receiver$ ls
gurupubkey.gpg  gurusaranpubkey.gpg  msgdecrypted  msgencrypted  msgtosendsigned
guru@ubuntu:~/Desktop/receiver$ gpg --verify msgtosendsigned
gpg: Signature made Thu 24 Sep 2020 11:08:38 AM IST
gpg:                using RSA key 59F65C0C0A07F0C0C3D5F2005C0C6F33537EDA809
gpg: Good signature from "venkat <gurusarand@gmail.com>" [unknown]
gpg: WARNING: This key is not certified with a trusted signature!
gpg:       There is no indication that the signature belongs to the owner.
Primary key fingerprint: 59F6 1EDC 2689 68ED 13DE  F290 5CD6 F33E 37ED A809
guru@ubuntu:~/Desktop/receiver$
  
```