Roll No: 09
Name: Shreya Bagade
Date: 08/09/2023

LAB ASSIGNMENT NO:13

Aim: Explore the GPG tool of linux to implement email security.

Lab Outcome Attained: Demonstrate the network security system using open source tools.

Theory:

1. What is private key ring and public key ring?

A ‘private keyring’ and a ‘public keyring’ typically refer to collections or sets of cryptographic keys, particularly in the context of public-key cryptography. These terms are often used when discussing secure communication, encryption, and digital signatures. Here's what each term means:

Private Keyring:

A private keyring is a collection of private keys associated with a user or entity.

Private keys are a fundamental component of public-key cryptography. They are secret keys that are kept confidential by the owner.

Private keys are used for tasks like decrypting data that has been encrypted with the corresponding public key, digitally signing documents, and proving the identity of the key's owner.

Protecting private keys is critical because if they are compromised, an attacker can impersonate the key's owner, decrypt encrypted data, and potentially cause security breaches.

Public Keyring:

A public keyring is a collection of public keys that are associated with users or entities and are made available to the public or other users.

Public keys are derived from private keys using mathematical algorithms and can be freely shared.

Public keys are used by others to encrypt data intended for the owner of the corresponding private key and to verify digital signatures generated by that private key.

Public keys are not confidential and can be distributed widely. They are a fundamental part of secure communication and identity verification in public-key cryptography systems.

2. Write the commands used for key generation, export and import of keys and signing and encrypting the message in gpg tool.

gpg --gen-key or gpg –full-generate-key

gpg --export -a username>filename (creates file in ascii format) or

gpg --output filename --armor --export user's_email

gpg --import their_public_key.asc Or gpg --import their_private_key.asc

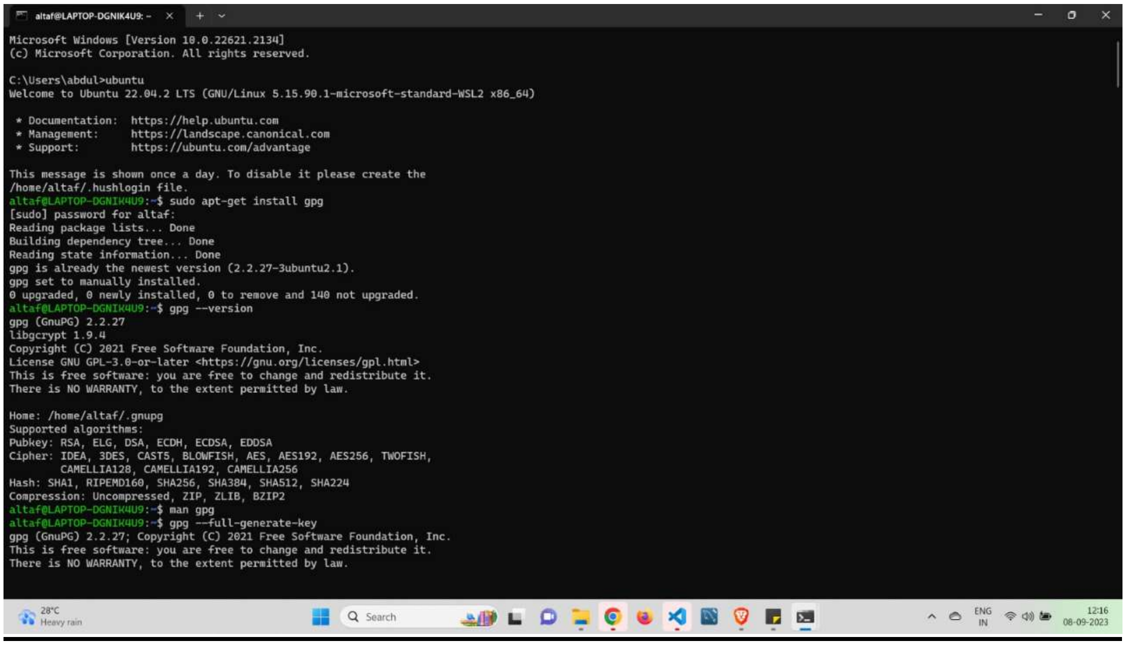
gpg --sign-key receiver_email

gpg --encrypt -r receiver_email name_of_file

gpg --encrypt --sign --armor -r receiver_email name_of_file OR

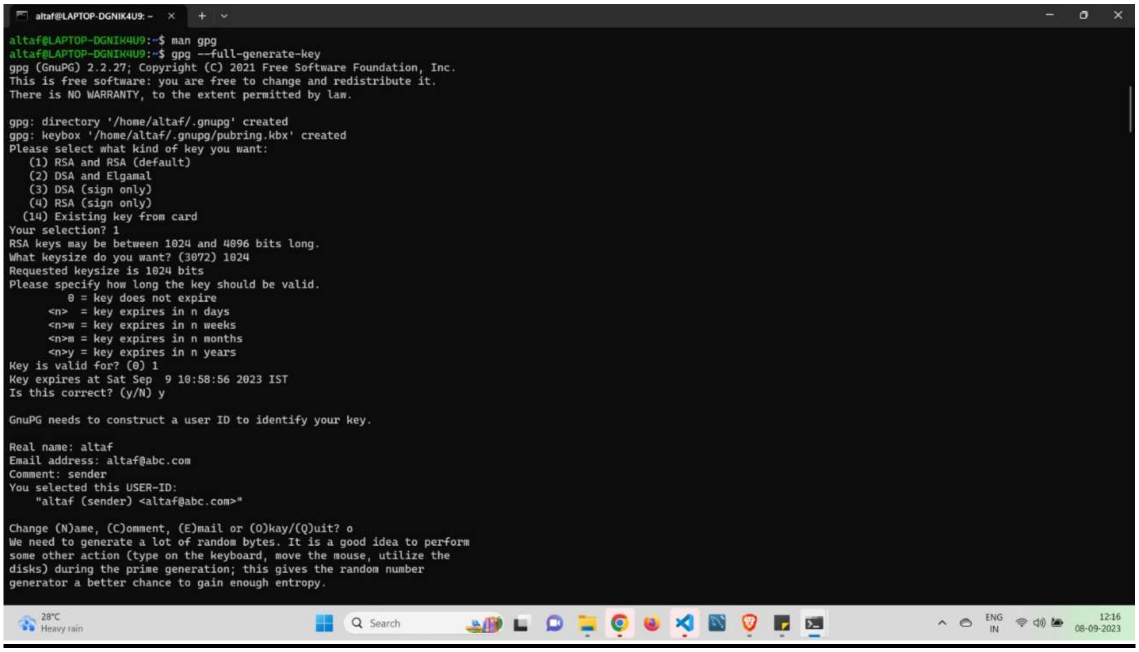
gpg --encrypt --sign -r receiver_email name_of_file

Output Screenshots:

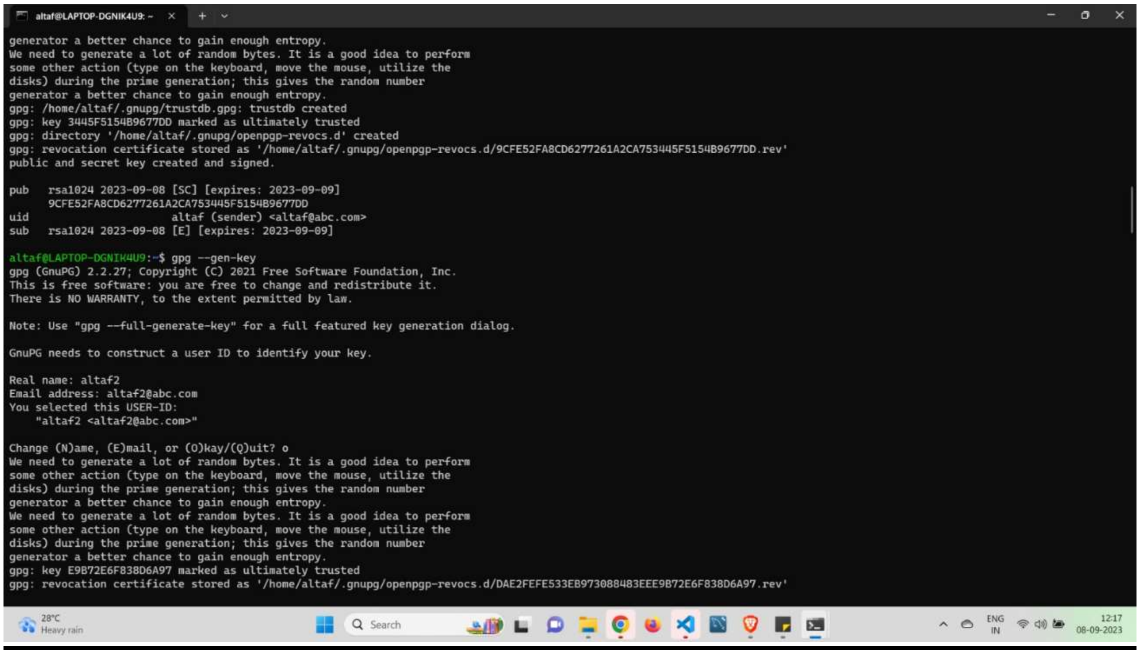


```
Microsoft Windows [Version 10.0.22621.2134]
(c) Microsoft Corporation. All rights reserved.

C:\Users\abdul>ubuntu
Welcome to Ubuntu 22.04.2 LTS (GNU/Linux 5.15.90.1-microsoft-standard-WSL2 x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

This message is shown once a day. To disable it please create the
/home/altaf/.hushlogin file.
altaf@LAPTOP-DGNIK4U9:~$ sudo apt-get install gpg
[sudo] password for altaf:
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
gpg is already the newest version (2.2.27-3ubuntu2.1).
gpg set to manually installed.
0 upgraded, 0 newly installed, 0 to remove and 140 not upgraded.
altaf@LAPTOP-DGNIK4U9:~$ gpg --version
gpg (GnuPG) 2.2.27
libgcrypt 1.9.4
Copyright (C) 2021 Free Software Foundation, Inc.
License GNU GPL-3.0-or-later <https://gnu.org/licenses/gpl.html>
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.

Home: /home/altaf/.gnupg
Supported algorithms:
Pubkey: RSA, ELG, DSA, ECDH, ECDSA, EDDSA
Cipher: IDEA, 3DES, CAST5, BLOWFISH, AES, AES192, AES256, TWOFISH,
        CAMELLIA128, CAMELLIA192, CAMELLIA256
Hash: SHA1, RIPEMD160, SHA256, SHA384, SHA512, SHA224
Compression: Uncompressed, ZIP, ZLIB, BZIP2
altaf@LAPTOP-DGNIK4U9:~$ man gpg
altaf@LAPTOP-DGNIK4U9:~$ gpg --full-generate-key
gpg (GnuPG) 2.2.27; Copyright (C) 2021 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.
```

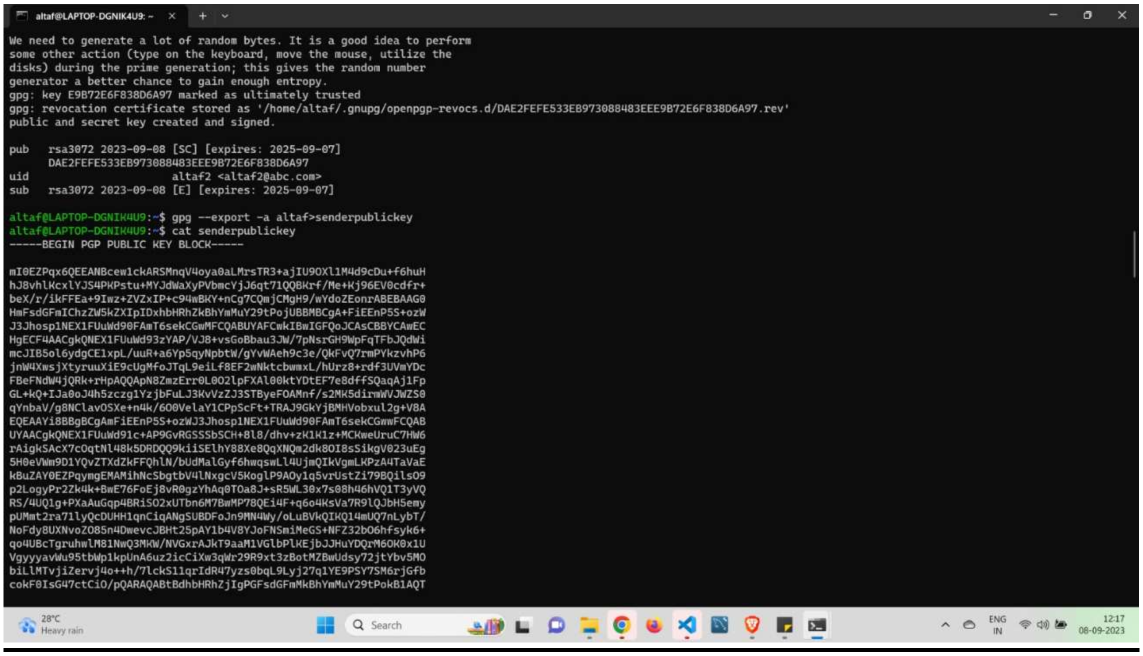
Roll No: 09
Name: Shreya Bagade
Date: 08/09/2023



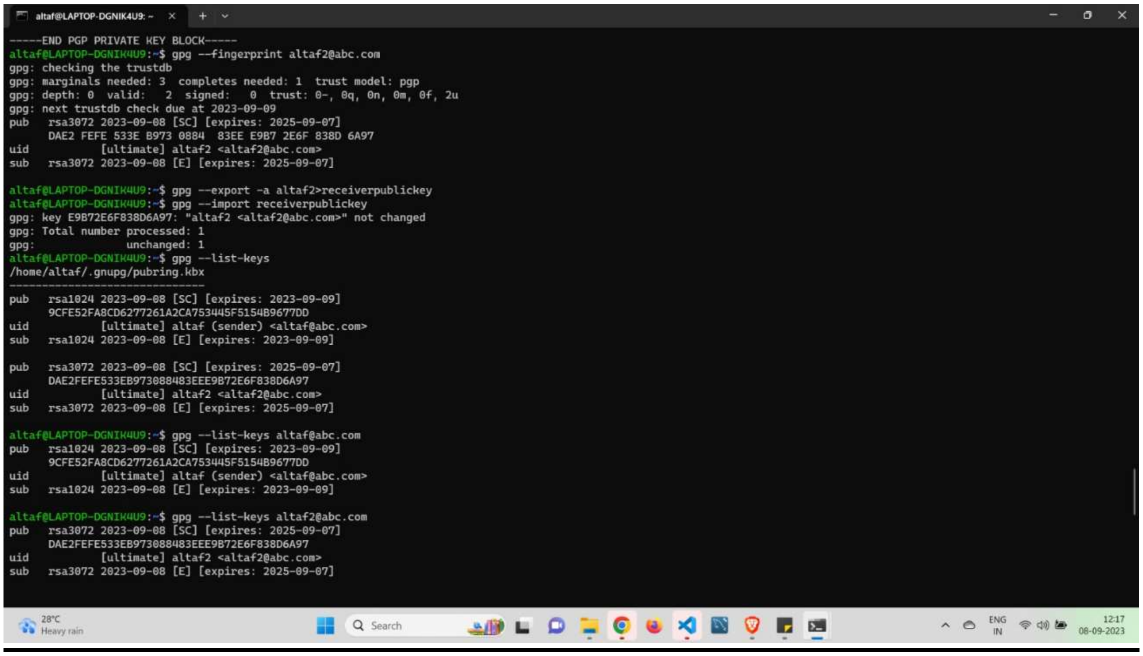
```
altaf@LAPTOP-DGNIK4U9:~$ man gpg
altaf@LAPTOP-DGNIK4U9:~$ gpg --full-generate-key
gpg (GnuPG) 2.2.27; Copyright (C) 2021 Free Software Foundation, Inc.
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.

gpg: directory '/home/altaf/.gnupg' created
gpg: keybox '/home/altaf/.gnupg/pubring.kbx' created
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)
  (14) Existing key from card
Your selection? 1
RSA keys may be between 1024 and 4096 bits long.
What keysize do you want? (3072) 1024
Requested keysize is 1024 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) 1
Key expires at Sat Sep  9 10:58:56 2023 IST
Is this correct? (y/N) y

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

Real name: altaf
Email address: altaf@abc.com
Comment: sender
You selected this USER-ID:
    "altaf (sender) <altaf@abc.com>"

Change (N)ame, (C)omment, (E)mail 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.
```



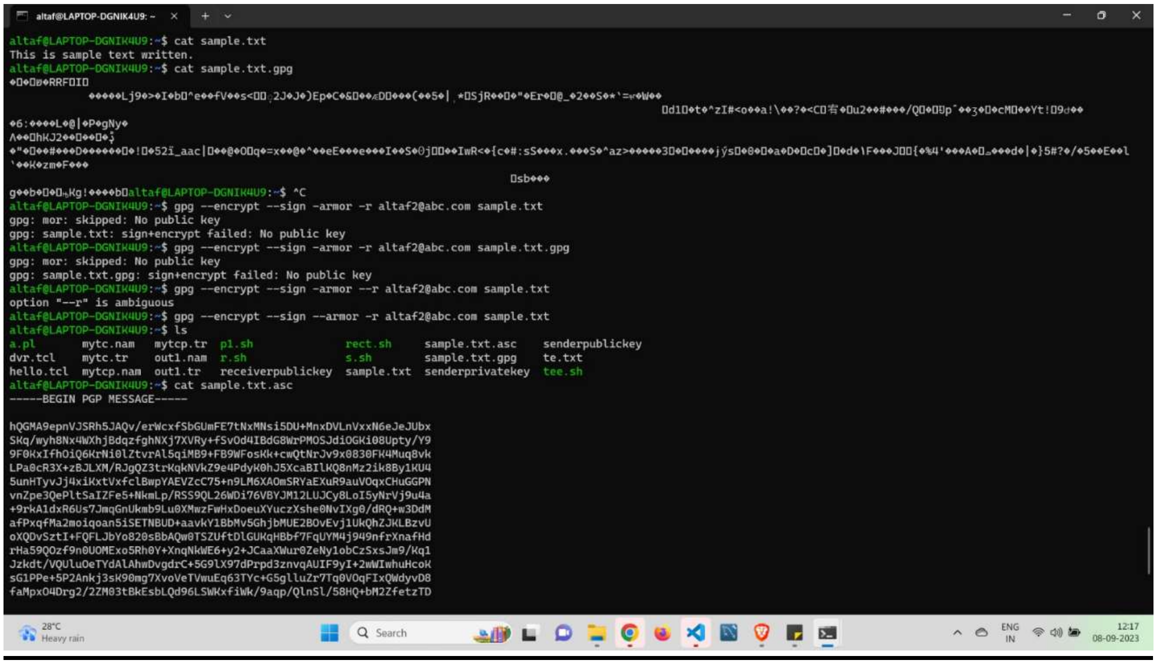
```
generator a better chance to gain enough entropy.
We need to generate a lot of random bytes. It is a good idea to perform
some other action (type on the keyboard, move the mouse, utilize the
disks) during the prime generation; this gives the random number
generator a better chance to gain enough entropy.
gpg: /home/altaf/.gnupg/trustdb.gpg: trustdb created
gpg: key 3445F5154B9677DD marked as ultimately trusted
gpg: directory '/home/altaf/.gnupg/openpgp-revocs.d' created
gpg: revocation certificate stored as '/home/altaf/.gnupg/openpgp-revocs.d/9CFE52FA8CD6277261A2CA753445F5154B9677DD.rev'
public and secret key created and signed.

pub   rsa1024 2023-09-08 [SC] [expires: 2023-09-09]
      9CFE52FA8CD6277261A2CA753445F5154B9677DD
uid                      altaf (sender) <altaf@abc.com>
sub   rsa1024 2023-09-08 [E] [expires: 2023-09-09]

altaf@LAPTOP-DGNIK4U9:~$ gpg --gen-key
gpg (GnuPG) 2.2.27; Copyright (C) 2021 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: altaf2
Email address: altaf2@abc.com
You selected this USER-ID:
    "altaf2 <altaf2@abc.com>"

Change (N)ame, (E)mail, 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 E9B72E6F838D6A97 marked as ultimately trusted
gpg: revocation certificate stored as '/home/altaf/.gnupg/openpgp-revocs.d/DAE2FEFE533EB973088483EEE9B72E6F838D6A97.rev'
```



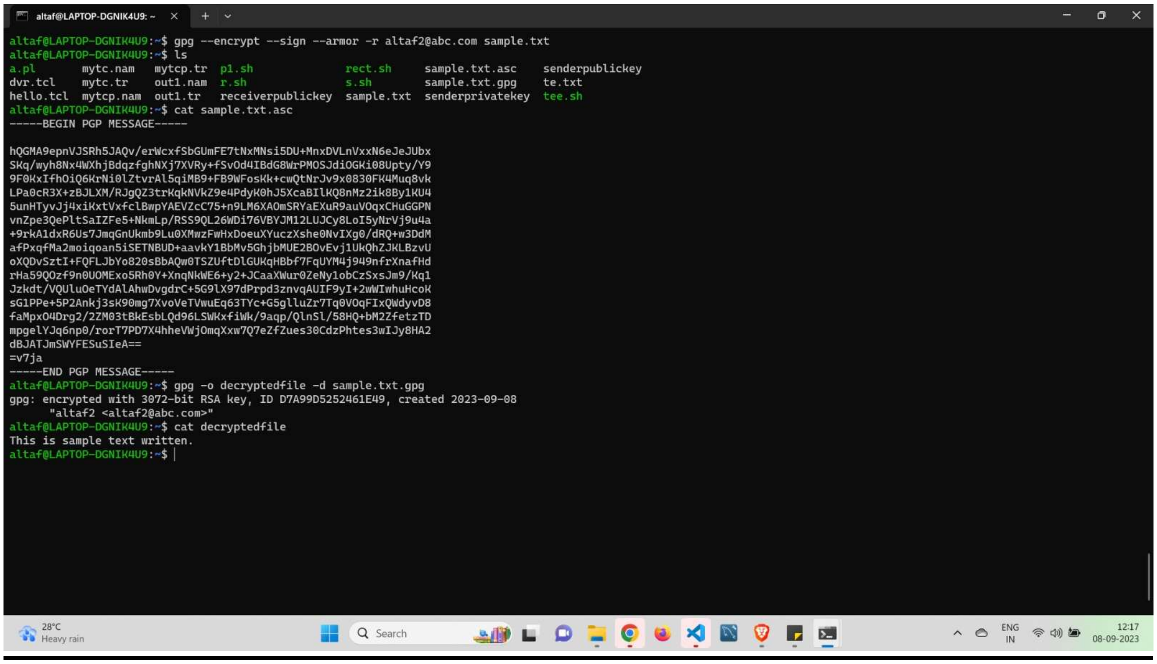
```
We need to generate a lot of random bytes. It is a good idea to perform
some other action (type on the keyboard, move the mouse, utilize the
disks) during the prime generation; this gives the random number
generator a better chance to gain enough entropy.
gpg: key E9B72E6F838D6A97 marked as ultimately trusted
gpg: revocation certificate stored as '/home/altaf/.gnupg/openpgp-revocs.d/DAE2FEFE533EB973088483EEE9B72E6F838D6A97.rev'
public and secret key created and signed.

pub   rsa3072 2023-09-08 [SC] [expires: 2025-09-07]
      DAE2FEFE533EB973088483EEE9B72E6F838D6A97
uid                      altaf2 <altaf2@abc.com>
sub   rsa3072 2023-09-08 [E] [expires: 2025-09-07]

altaf@LAPTOP-DGNIK4U9:~$ gpg --export -a altaf>senderpublickey
altaf@LAPTOP-DGNIK4U9:~$ cat senderpublickey
-----BEGIN PGP PUBLIC KEY BLOCK-----

mI0EZPqx6QEEANBcew1ckARSMnqV4oya0aLMrsTR3+ajIU9OXl1M4d9cDu+f6huH
hJ8vhlKcxlYJS4PKPstu+MYJdWaXyPVbmcYjJ6qt71QQBKrf/Me+Kj96EV0cdfr+
beX/r/ikFFEa+9Iwz+ZVZxIP+c94wBKY+nCg7CQmjCMgH9/wYdoZEonrABEBAAG0
HmFsdGFmIChzZW5kZXIpIDxhbHRhZkBhYmMuY29tPojUBBMBCgA+FiEEnP5S+ozW
J3Jhosp1NEX1FUuWd90FAmT6sekCGwMFCQABUYAFCwkIBwIGFQoJCAsCBBYCAwEC
HgECF4AACgkQNEX1FUuWd93zYAP/VJ8+vsGoBbau3JW/7pNsrGH9WpFqTFbJQdWi
mcJIB5ol6ydgCE1xpL/uuR+a6Yp5qyNpbtW/gYvWAeh9c3e/QkFvQ7rmPYkzvhP6
jnW4XwsjXtyruuXiE9cUgMfoJTqL9eiLf8EF2wNktcbwmxL/hUrz8+rdf3UVmYDc
FBeFNdW4jQRk+rHpAQQApN8ZmzErr0L0O2lpFXAl00ktYDtEF7e8dffSQaqAj1Fp
GL+kQ+IJa0oJ4h5zczg1YzjbFuLJ3KvVzZJ3STByeFOAMnf/s2MK5dirmWVJWZS0
qYnbaV/g8NClavOSXe+n4k/6O0VelaY1CPpScFt+TRAJ9GkYjBMHVobxul2g+V8A
EQEAAYi8BBgBCgAmFiEEnP5S+ozWJ3Jhosp1NEX1FUuWd90FAmT6sekCGwwFCQAB
UYAACgkQNEX1FUuWd91c+AP9GvRGSSSbSCH+8l8/dhv+zK1K1z+MCKweUruC7HW6
rAigkSAcX7cOqtNl48k5DRDQQ9kiiSElhY88Xe8QqXNQm2dk8OI8sSikgV023uEg
5H0eVWm9D1YQvZTXdZkFFQhlN/bUdMalGyf6hwqswLl4UjmQIkVgmLKPzA4TaVaE
kBuZAY0EZPqymgEMAMihNcSbgtbV4lNxgcV5KoglP9AOy1q5vrUstZi79BQilsO9
p2LogyPr2Zk4k+BwE76FoEj8vR0gzYhAq0TOa8J+sR5WL30x7s08h46hVQ1T3yVQ
RS/4UQ1g+PXaAuGqp4BRiSO2xUTbn6M7BwMP78QEi4F+q6o4KsVa7R9lQJbH5emy
pUMmt2ra71lyQcDUHH1qnCiqANgSUBDFoJn9MN4Wy/oLuBVkQIKQ14mUQ7nLybT/
NoFdy8UXNvoZO85n4DwevcJBHt25pAY1b4V8YJoFNSmiMeGS+NFZ32bO6hfsyk6+
qo4UBcTgruhwlM81NwQ3MKW/NVGxrAJkT9aaM1VGlbPlKEjbJJHuYDQrM6OK0x1U
VgyyyavWu95tbWp1kpUnA6uz2icCiXw3qWr29R9xt3zBotMZBwUdsy72jtYbv5MO
biLlMTvjiZervj4o++h/7lckS11qrIdR47yzs0bqL9Lyj27q1YE9PSY7SM6rjGfb
cokF0IsG47ctCiO/pQARAQABtBdhbHRhZjIgPGFsdGFmMkBhYmMuY29tPokB1AQT
```



```
-----END PGP PRIVATE KEY BLOCK-----
altaf@LAPTOP-DGNIK4U9:~$ gpg --fingerprint altaf2@abc.com
gpg: checking the trustdb
gpg: marginals needed: 3  completes needed: 1  trust model: pgp
gpg: depth: 0  valid:   2  signed:   0  trust: 0-, 0q, 0n, 0m, 0f, 2u
gpg: next trustdb check due at 2023-09-09
pub   rsa3072 2023-09-08 [SC] [expires: 2025-09-07]
      DAE2 FEFE 533E B973 0884  83EE E9B7 2E6F 838D 6A97
uid           [ultimate] altaf2 <altaf2@abc.com>
sub   rsa3072 2023-09-08 [E] [expires: 2025-09-07]

altaf@LAPTOP-DGNIK4U9:~$ gpg --export -a altaf2>receiverpublickey
altaf@LAPTOP-DGNIK4U9:~$ gpg --import receiverpublickey
gpg: key E9B72E6F838D6A97: "altaf2 <altaf2@abc.com>" not changed
gpg: Total number processed: 1
gpg:              unchanged: 1
altaf@LAPTOP-DGNIK4U9:~$ gpg --list-keys
/home/altaf/.gnupg/pubring.kbx
------------------------------
pub   rsa1024 2023-09-08 [SC] [expires: 2023-09-09]
      9CFE52FA8CD6277261A2CA753445F5154B9677DD
uid           [ultimate] altaf (sender) <altaf@abc.com>
sub   rsa1024 2023-09-08 [E] [expires: 2023-09-09]

pub   rsa3072 2023-09-08 [SC] [expires: 2025-09-07]
      DAE2FEFE533EB973088483EEE9B72E6F838D6A97
uid           [ultimate] altaf2 <altaf2@abc.com>
sub   rsa3072 2023-09-08 [E] [expires: 2025-09-07]

altaf@LAPTOP-DGNIK4U9:~$ gpg --list-keys altaf@abc.com
pub   rsa1024 2023-09-08 [SC] [expires: 2023-09-09]
      9CFE52FA8CD6277261A2CA753445F5154B9677DD
uid           [ultimate] altaf (sender) <altaf@abc.com>
sub   rsa1024 2023-09-08 [E] [expires: 2023-09-09]

altaf@LAPTOP-DGNIK4U9:~$ gpg --list-keys altaf2@abc.com
pub   rsa3072 2023-09-08 [SC] [expires: 2025-09-07]
      DAE2FEFE533EB973088483EEE9B72E6F838D6A97
uid           [ultimate] altaf2 <altaf2@abc.com>
sub   rsa3072 2023-09-08 [E] [expires: 2025-09-07]
```



```
altaf@LAPTOP-DGNIK4U9:~$ cat sample.txt
This is sample text written.
altaf@LAPTOP-DGNIK4U9:~$ cat sample.txt.gpg
◆□◆□Ø◆RRF□I□
        ◆◆◆◆◆Lj9◆>◆I◆b□^e◆◆fV◆◆s<□□ 2J◆J◆)Ep◆C◆&□◆◆ꬱD□◆◆◆(◆◆5◆| *□SjR◆◆□◆"◆Er◆□@_◆2◆◆S◆*'=ᵣr◆W◆◆
                                                                      □d1□◆t◆^zI#<o◆◆a!\◆◆?◆<C□宥◆□u2◆◆#◆◆◆/Q□◆□𝔍p˘◆◆ვ◆□◆cM□◆◆Yt!□9d◆◆
◆6:◆◆◆◆L◆@|◆P◆gNy◆
Λ◆◆□hKJ2◆◆□◆◆□◆ȷ̈
◆"◆□◆◆#◆◆◆D◆◆◆◆◆◆□◆!□◆52ĭ_aac|□◆◆@◆O□q◆=x◆◆@◆^◆◆eE◆◆◆e◆◆◆I◆◆S◆0j□□◆◆IwR<◆{c◆#:sS◆◆◆x.◆◆◆S◆^az>◆◆◆◆◆3□◆□◆◆◆◆jýs□◆0◆□◆a◆D◆□c□◆]□◆d◆\F◆◆◆J□□{◆%4'◆◆◆A◆□ₒ◆◆◆d◆|◆}5#?◆/◆5◆◆E◆◆l
'◆◆K◆zm◆F◆◆◆
                                                     □sb◆◆◆
g◆◆b◆□◆□ₕKg!◆◆◆◆b□altaf@LAPTOP-DGNIK4U9:~$ ^C
altaf@LAPTOP-DGNIK4U9:~$ gpg --encrypt --sign -armor -r altaf2@abc.com sample.txt
gpg: mor: skipped: No public key
gpg: sample.txt: sign+encrypt failed: No public key
altaf@LAPTOP-DGNIK4U9:~$ gpg --encrypt --sign -armor -r altaf2@abc.com sample.txt.gpg
gpg: mor: skipped: No public key
gpg: sample.txt.gpg: sign+encrypt failed: No public key
altaf@LAPTOP-DGNIK4U9:~$ gpg --encrypt --sign -armor --r altaf2@abc.com sample.txt
option "--r" is ambiguous
altaf@LAPTOP-DGNIK4U9:~$ gpg --encrypt --sign --armor -r altaf2@abc.com sample.txt
altaf@LAPTOP-DGNIK4U9:~$ ls
a.pl       mytc.nam   mytcp.tr  p1.sh               rect.sh     sample.txt.asc    senderpublickey
dvr.tcl    mytc.tr    out1.nam  r.sh                s.sh        sample.txt.gpg    te.txt
hello.tcl  mytcp.nam  out1.tr   receiverpublickey   sample.txt  senderprivatekey  tee.sh
altaf@LAPTOP-DGNIK4U9:~$ cat sample.txt.asc
-----BEGIN PGP MESSAGE-----

hQGMA9epnVJSRh5JAQv/erWcxfSbGUmFE7tNxMNsi5DU+MnxDVLnVxxN6eJeJUbx
SKq/wyh8Nx4WXhjBdqzfghNXj7XVRy+fSvOd4IBdG8WrPMOSJdiOGKi08Upty/Y9
9F0KxIfhOiQ6KrNi0lZtvrAl5qiMB9+FB9WFosKk+cwQtNrJv9x0830FK4Muq8vk
LPa0cR3X+zBJLXM/RJgQZ3trKqkNVkZ9e4PdyK0hJ5XcaBIlKQ8nMz2ik8By1KU4
5unHTyvJj4xiKxtVxfclBwpYAEVZcC75+n9LM6XAOmSRYaEXuR9auVOqxCHuGGPN
vnZpe3QePltSaIZFe5+NkmLp/RSS9QL26WDi76VBYJM12LUJCy8LoI5yNrVj9u4a
+9rkA1dxR6Us7JmqGnUkmb9Lu0XMwzFwHxDoeuXYuczXshe0NvIXg0/dRQ+w3DdM
afPxqfMa2moiqoan5iSETNBUD+aavkY1BbMv5GhjbMUE2BOvEvj1UkQhZJKLBzvU
oXQDvSztI+FQFLJbYo820sBbAQw0TSZUftDlGUKqHBbf7FqUYM4j949nfrXnafHd
rHa59QOzf9n0UOMExo5Rh0Y+XnqNkWE6+y2+JCaaXWur0ZeNy1obCzSxsJm9/Kq1
Jzkdt/VQUluOeTYdAlAhwDvgdrC+5G9lX97dPrpd3znvqAUIF9yI+2wWIwhuHcoK
sG1PPe+5P2Ankj3sK90mg7XvoVeTVwuEq63TYc+G5glluZr7Tq0VOqFIxQWdyvD8
faMpxO4Drg2/2ZM03tBkEsbLQd96LSWKxfiWk/9aqp/QlnSl/58HQ+bM2ZfetzTD
```



```
altaf@LAPTOP-DGNIK4U9:~$ gpg --encrypt --sign --armor -r altaf2@abc.com sample.txt
altaf@LAPTOP-DGNIK4U9:~$ ls
a.pl       mytc.nam   mytcp.tr  p1.sh               rect.sh     sample.txt.asc    senderpublickey
dvr.tcl    mytc.tr    out1.nam  r.sh                s.sh        sample.txt.gpg    te.txt
hello.tcl  mytcp.nam  out1.tr   receiverpublickey   sample.txt  senderprivatekey  tee.sh
altaf@LAPTOP-DGNIK4U9:~$ cat sample.txt.asc
-----BEGIN PGP MESSAGE-----

hQGMA9epnVJSRh5JAQv/erWcxfSbGUmFE7tNxMNsi5DU+MnxDVLnVxxN6eJeJUbx
SKq/wyh8Nx4WXhjBdqzfghNXj7XVRy+fSvOd4IBdG8WrPMOSJdiOGKi08Upty/Y9
9F0KxIfhOiQ6KrNi0lZtvrAl5qiMB9+FB9WFosKk+cwQtNrJv9x0830FK4Muq8vk
LPa0cR3X+zBJLXM/RJgQZ3trKqkNVkZ9e4PdyK0hJ5XcaBIlKQ8nMz2ik8By1KU4
5unHTyvJj4xiKxtVxfclBwpYAEVZcC75+n9LM6XAOmSRYaEXuR9auVOqxCHuGGPN
vnZpe3QePltSaIZFe5+NkmLp/RSS9QL26WDi76VBYJM12LUJCy8LoI5yNrVj9u4a
+9rkA1dxR6Us7JmqGnUkmb9Lu0XMwzFwHxDoeuXYuczXshe0NvIXg0/dRQ+w3DdM
afPxqfMa2moiqoan5iSETNBUD+aavkY1BbMv5GhjbMUE2BOvEvj1UkQhZJKLBzvU
oXQDvSztI+FQFLJbYo820sBbAQw0TSZUftDlGUKqHBbf7FqUYM4j949nfrXnafHd
rHa59QOzf9n0UOMExo5Rh0Y+XnqNkWE6+y2+JCaaXWur0ZeNy1obCzSxsJm9/Kq1
Jzkdt/VQUluOeTYdAlAhwDvgdrC+5G9lX97dPrpd3znvqAUIF9yI+2wWIwhuHcoK
sG1PPe+5P2Ankj3sK90mg7XvoVeTVwuEq63TYc+G5glluZr7Tq0VOqFIxQWdyvD8
faMpxO4Drg2/2ZM03tBkEsbLQd96LSWKxfiWk/9aqp/QlnSl/58HQ+bM2ZfetzTD
mpgelYJq6np0/rorT7PD7X4hheVWjOmqXxw7Q7eZfZues30CdzPhtes3wIJy8HA2
dBJATJmSWYFESuSIeA==
=v7ja
-----END PGP MESSAGE-----
altaf@LAPTOP-DGNIK4U9:~$ gpg -o decryptedfile -d sample.txt.gpg
gpg: encrypted with 3072-bit RSA key, ID D7A99D5252461E49, created 2023-09-08
      "altaf2 <altaf2@abc.com>"
altaf@LAPTOP-DGNIK4U9:~$ cat decryptedfile
This is sample text written.
altaf@LAPTOP-DGNIK4U9:~$ |
```

<u>Conclusion:</u> This experiment with the GnuPG tool for PGP (Pretty Good Privacy) has demonstrated its effectiveness in securing digital communication through encryption and digital signatures. We successfully generated key pairs, encrypted and decrypted messages, and verified digital signatures. This powerful tool provides a robust framework for ensuring the confidentiality, integrity, and authenticity of sensitive data in the realm of secure communication.