

# Lab Project on using PGP – GNU Privacy Guard (GPG) for Secure E-mail Communication

Prepared by: **Dr. Natarajan Meghanathan**

## Pretty Good Privacy (PGP)

PGP is a secure E-mail communication standard that provides cryptographic privacy and authentication. PGP supports both symmetric keys-based and public/private key pairs-based secure communication between communicating parties. In this project, we will do both symmetric as well as public key-based encryption. We will use GNU Privacy Guard (GPG) – an open-source version of PGP in our project.

**WHAT TO SUBMIT:** Read through the entire project description and follow the steps as indicated in detail. Take screenshots of Figures 1 through 24 that get generated as you do the project by yourself according to the requirements indicated. Submit your detailed project report with Figures 1 through 24 clearly labeled. Also explain the working of the PGP for both symmetric and public-key encryptions using a flowchart, for each case, indicating the sequence of steps, in a nutshell.

## Stage 1: Installing GPG on your KUbuntu Virtual Machine

This project would continue from the previous project for which you installed the KUbuntu (or a similar variant) virtual machine (VM) on VMware Player.

1. Login to the KUbuntu VM using the username (vmplanet) and password (vmplanet.net).
2. Open a Konsole terminal by searching for “terminal” after clicking the K-button on the bottom left of the VMware Player screen.
3. In the Konsole terminal, run the command “sudo apt-get install pgpgpg”. When asked for the password, enter *vmplanet.net*. The GPG installation will start and wait for it to complete.
4. Following step 3, run the command “sudo apt-get install gnupg-agent” in the Konsole terminal. When asked for password, enter *vmplanet.net*. The GnuPG-Agent installation will start and wait for it to complete.

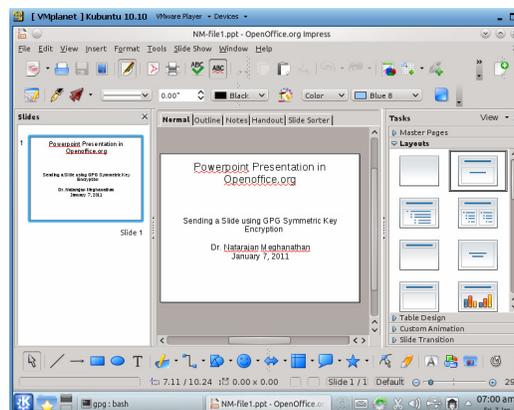
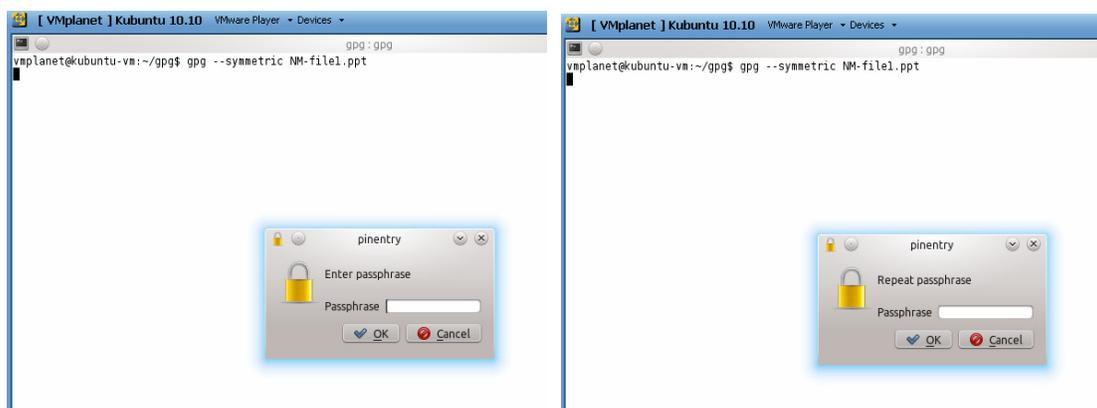


Figure 1: Powerpoint Presentation in Open Office

## Stage 2: Symmetric Key Encryption/ Decryption with GPG

In this phase of the project, we will create a Powerpoint (.ppt) file using the Open Office applications available in KUbuntu. We are going to encrypt the .ppt file at one end using GPG and a symmetric key and send it as a secure attachment over e-mail to the other end where we will decrypt using GPG and the same symmetric key, extract and view the .ppt file.

1. From the Applications Menu (obtained by clicking the 'K' button at the bottom of the VMware Player screen, open a presentation (open office.org).
2. Type the contents in a slide, something similar to the one shown in Figure 1, with your name and date appearing on it. Save the file as a Microsoft 97/2000/XP compatible .ppt format. Note down the location where you save the file. In this project description, I am storing it in the folder /home/vmplanet/gpg. You can cross check by going to the Konsole terminal and to the /home/vmplanet/gpg folder and make sure the .ppt file is there. I have named the .ppt file as NM-file1.ppt, where 'N' and 'M' are the first characters of my first name and last name respectively. You should also follow a similar convention, based on your first name and last name, while doing your project.  
 Note: By default, you would be logged into the /home/vmplanet/ folder. You may want to create a 'gpg' folder for yourself to do this project, though this is not mandatory. You can save your file anywhere insider any folder.
3. Now, run the command "gpg --symmetric NM-file1.ppt". You will be prompted for a passphrase and to repeat what you entered again. In this project description, I used *secprj* as the passphrase. You are free to use anything that you wish. But, remember that passphrase as you will need it to decrypt. Now run the `ls -l` command on your terminal, you should be able to see a .gpg file. In Figure 2, it appears as NM-file1.ppt.gpg. If the 'pinentry' window does not popup, click on it at the toolbar in the bottom, it will popup.



```

vmplanet@kubuntu-vm:~/gpg$ gpg --symmetric NM-file1.ppt
vmplanet@kubuntu-vm:~/gpg$ ls -l
total 80
-rw-r--r-- 1 vmplanet vmplanet 72192 2011-01-07 06:47 NM-file1.ppt
-rw-r--r-- 1 vmplanet vmplanet 6604 2011-01-07 06:50 NM-file1.ppt.gpg
vmplanet@kubuntu-vm:~/gpg$ █
  
```

Figure 2: GPG Symmetric Key Encryption of the Powerpoint File

```

vmplanet@kubuntu-vm:~/gpg$ cp NM-file1.ppt.gpg email-NM-file1.ppt.gpg
vmplanet@kubuntu-vm:~/gpg$ ls -l
total 88
-rw-r--r-- 1 vmplanet vmplanet 6604 2011-01-07 06:53 email-NM-file1.ppt.gpg
-rw-r--r-- 1 vmplanet vmplanet 72192 2011-01-07 06:47 NM-file1.ppt
-rw-r--r-- 1 vmplanet vmplanet 6604 2011-01-07 06:50 NM-file1.ppt.gpg
vmplanet@kubuntu-vm:~/gpg$ █
  
```

Figure 3: Copying the encrypted GPG file and saving as another file

4. You can actually e-mail this NM-file1.ppt.gpg file as an attachment and it can be downloaded and saved at the receiver side in a different file name, but of course with the “.ppt.gpg” extension. In this project description, I have just copied the NM-file1.ppt.gpg file to another file with name email-NM-file1.ppt.gpg in the same folder. You can use the “cp” command as I have shown. Use ls -l to see if the new file is there. See Figure 3 above.
5. Run the command “gpg --output extracted-NM-file1.ppt -d email-NM-file1.ppt.gpg” as shown in Figure 4, you will be prompted for the passphrase. The extracted contents will be in the extracted-NM-file1.ppt file. You can cross-check by opening the extracted .ppt file.

```
vmplanet@kubuntu-vm:~/gpg$ gpg --output extracted-NM-file1.ppt -d email-NM-file1.ppt.gpg
gpg: CAST5 encrypted data
gpg: encrypted with 1 passphrase
gpg: WARNING: message was not integrity protected
vmplanet@kubuntu-vm:~/gpg$ ls -l
total 160
-rw-r--r-- 1 vmplanet vmplanet 6604 2011-01-07 06:53 email-NM-file1.ppt.gpg
-rw-r--r-- 1 vmplanet vmplanet 72192 2011-01-07 06:59 extracted-NM-file1.ppt
-rw-r--r-- 1 vmplanet vmplanet 72192 2011-01-07 06:47 NM-file1.ppt
-rw-r--r-- 1 vmplanet vmplanet 6604 2011-01-07 06:50 NM-file1.ppt.gpg
vmplanet@kubuntu-vm:~/gpg$ █
```

Figure 4: Decrypting and extracting the Powerpoint file

```
vmplanet@kubuntu-vm:~/gpg$ gpg --symmetric --armor NM-file1.ppt
vmplanet@kubuntu-vm:~/gpg$ ls -l
total 172
-rw-r--r-- 1 vmplanet vmplanet 6604 2011-01-07 06:53 email-NM-file1.ppt.gpg
-rw-r--r-- 1 vmplanet vmplanet 72192 2011-01-07 06:59 extracted-NM-file1.ppt
-rw-r--r-- 1 vmplanet vmplanet 72192 2011-01-07 06:47 NM-file1.ppt
-rw-r--r-- 1 vmplanet vmplanet 9042 2011-01-08 01:36 NM-file1.ppt.asc
-rw-r--r-- 1 vmplanet vmplanet 6604 2011-01-07 06:50 NM-file1.ppt.gpg
vmplanet@kubuntu-vm:~/gpg$ █
```

Figure 5: Encrypting the Powerpoint to printable ASCII characters

6. Now, instead of attaching the encrypted .gpg file, if you want to send the encrypted data as part of the e-mail body, then you can encrypt the NM-file1.ppt using the command: “gpg --symmetric --armor NM-file1.ppt”. You will be prompted to enter a passphrase and repeat it. You can enter a different passphrase or the same passphrase that you used in Step 3. But, remember the passphrase you entered now. An encrypted file by the name NM-file1.ppt.asc would have been created.
7. A run of the ls -l command will list the NM-file1.ppt.asc (see Figure 5) and if you run the cat command as: “cat NM-file1.ppt.asc”, a long list of printable ASCII characters would be displayed (see Figure 6). You can open your email compose box and copy and paste the contents of the NM-file1.ppt.asc and send it to the receiver. Only with gpg and the correct passphrase, someone can decrypt the message.

```

vmplanet@kubuntu-vm:~/gpg$ cat NM-file1.ppt.asc
-----BEGIN PGP MESSAGE-----
Version: GnuPG v1.4.10 (GNU/Linux)

jA0EAWMcpe6Q6GHU4VhgyezTkmjMTYpzLjZRV8r0cKiL2wFK2xUYUSa0Zlc3y+lH
GfRGsPC+tpWtK4MgQd0Kc4Yz2En/Ymtv9plT0oiLTrd18fg/1Du1lo7KMnw3tGAZ
2ZMfI+nXz5xg1YLNi2mKnVEYPnoB0yQtswdcTMX4UCYBsdu77aXZnNk0QggsTvLY
LSeR+Z0MdjSKT4HdQhj2725xk5ftDSArgx1BfvQGpdmIo5sUBoZPzsF028No0Nj/
qhRm0TgFQGi2MP05xE02Uoq0qqL+VVsQmMgwaL+hJ73mdqGk lkFnaMyN9Z2nC7
qNm5Nav2X/w1MX/tWbrg1WslVcTf8ELtmrakGdTfQme4oe9xaE2c8VV/Fsh0zbbj
xfIfV7T2/tf/Z1njptchyeinu8jZ2b0KvO/EePGXQgXZC80o0CaLRkxm99Ri iTX
X2os3NpdvORmndwLcweoSoRoBvZyb+3RwBN6z13DpIMq2IX9vrz93X+GIKXJrW8e
cFjk4WaZhH6NdbF7/zBoM20R8hyN2WVG2p7w1DQ/F7HJDN/bKaFslwzyNlqxRiTR
arVU3u4xVJtIrL5cA3QJ029Ht9BMMyxAjPD8JGS+pcRSSfgcVbySYCnpVDIw67V8
zwFRKR7ab5YpB1/KzTx4oJm9uZfeqKXBIYEdJFnkQhV34H0KEPLhk63wpvLaFiV
8sIe/Axv8gsEMRGUAGorBuft11b534o0l+CmJzB0rm0w8hsBtUaRGFudjyGW5P5i
I10Ta8EDK5BuiKJSOZNaPtVCv9ieYOKsZBiyyFy6Lz7qWVTNniK5/LA4cI2hL/j/
MP/rWKQzCMQAmqiQEr2Y9RaQLmUokZV4nRm7qVXN7EeOpIqIs0s0BcHieCX9bL6C
aIae94jdd207zeymYv4J8aLlzQEidqP33Sj/mMXztneXm9+NbNnd0U1a6ehqHe3E
byf/dtDdJNJ+7UwghQW3C1YA8+uekTBcHUQsFx8dLkF4weITgQAhoSNLZ4AkhlI
wrXld6Utqq6MFjmypb9LRuU/ev0YrsvRnbPG50o7EYgb1/tTw0BgtjUT6u92vLY
jjXIRlqMhyvszpuD75wHYJEX2yIA5fZhg3iTwEWxtGL6YGmK7thlvSRcfFcS7S
PBXjgmD92XMoBTWuu4xsCU/P9FNNRp3+fnKrpHMTz/xRJAg78-1YvqNx+E+cQM
PRifvLwKIJB40bkeS2x5jEBAZLGFf56xV8qeL8fnAALjZVrGvhY8W8Fjhzy0vopA
pS4aZcKTUirYqYgGB/udPHQSAW11F6EjyqimMeiZe431rnCUR6rAJoFku9KqEoTE
QfoeI26rkrXb9QXqsbd8aEzEzLnr4/FEoQ8CtYaQUlTtTjRorTphV/TEv80t5z

```

Figure 6: A portion of the encrypted PGP message of the Powerpoint file

- To decrypt the encrypted file NM-file1.ppt.asc, run the gpg command as: “gpg --output ext-NM-file1.ppt --armor -d NM-file1.ppt.asc”. The extracted Powerpoint file would be stored with a file name ext-NM-file1.ppt. You can use the ls -l command to check whether the file has been extracted and is available in the folder.

```

vmplanet@kubuntu-vm:~/gpg$ ls
email-NM-file1.ppt.gpg  extracted-NM-file1.ppt  NM-file1.ppt  NM-file1.ppt.asc  NM-file1.ppt.gpg
vmplanet@kubuntu-vm:~/gpg$ gpg --output ext-NM-file1.ppt --armor -d NM-file1.ppt.asc
gpg: CAST5 encrypted data
gpg: encrypted with 1 passphrase
gpg: WARNING: message was not integrity protected
vmplanet@kubuntu-vm:~/gpg$ ls -l
total 244
-rw-r--r-- 1 vmplanet vmplanet 6604 2011-01-07 06:53 email-NM-file1.ppt.gpg
-rw-r--r-- 1 vmplanet vmplanet 72192 2011-01-08 02:39 ext-NM-file1.ppt
-rw-r--r-- 1 vmplanet vmplanet 72192 2011-01-07 06:59 extracted-NM-file1.ppt
-rw-r--r-- 1 vmplanet vmplanet 72192 2011-01-07 06:47 NM-file1.ppt
-rw-r--r-- 1 vmplanet vmplanet 9042 2011-01-08 01:36 NM-file1.ppt.asc
-rw-r--r-- 1 vmplanet vmplanet 6604 2011-01-07 06:50 NM-file1.ppt.gpg
vmplanet@kubuntu-vm:~/gpg$ █

```

Note the difference in the file sizes

Figure 8: Extraction of the plaintext Powerpoint file from the ciphertext ASCII character file

### Stage 3: Public Key Encryption/ Decryption with GPG

Scenario: The scenario here is there are two users: *nmeghanathan* (referred to as the ‘sender’) and *natarajan\_jsu* (referred to as the ‘receiver’), with e-mail accounts [nmeghanathan@jsums.edu](mailto:nmeghanathan@jsums.edu) and [natarajan\\_jsu@hotmail.com](mailto:natarajan_jsu@hotmail.com) respectively. We are going to create new user accounts with usernames *nmeghanathan* and *natarajan\_jsu* in our KUbuntu machine. We are then going to create the public-key and private-key pair for the two users. The receiver account exports its public key to the sender and the latter uses it to encrypt a secret message and send the encrypted version back to the receiver. Now, the receiver decrypts the ciphertext message with its private key and extracts the message. In this scenario, we are going to send a simple text file as our message. The details are explained below:

Accounts to be created by the students: Following the analogy described here, you should create two user accounts that somehow capture your first name and/or last name and that you are associated with JSU.

Associate your JSU email address with the sender account and your non-JSU address (any active email address should work) with the receiver account.

1. In this projection description, I create two user accounts by name, nmeghanathan and natarajan\_jsu.

Figure 9 illustrates the user account creation step.

Note: It does not matter from which folder you create the two user accounts. I just create it from the /home/vmplanet folder. Note that you may also be prompted to enter the VM password vmplanet.net the first time you try to create an account. Then, for each of the two accounts, you will be asked to enter a UNIX password and reconfirm it. Choose a password of your choice for each of the two user accounts.

```
vmplanet@kubuntu-vm:~$ sudo adduser nmeghanathan
Adding user `nmeghanathan' ...
Adding new group `nmeghanathan' (1005) ...
Adding new user `nmeghanathan' (1005) with group `nmeghanathan' ...
Creating home directory `/home/nmeghanathan' ...
Copying files from `/etc/skel' ...
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
Changing the user information for nmeghanathan
Enter the new value, or press ENTER for the default
    Full Name []: Natarajan Meghanathan
    Room Number []:
    Work Phone []:
    Home Phone []:
    Other []:
Is the information correct? [Y/n] y
vmplanet@kubuntu-vm:~$
```

```
vmplanet@kubuntu-vm:~$ sudo adduser natarajan_jsu
Adding user `natarajan_jsu' ...
Adding new group `natarajan_jsu' (1006) ...
Adding new user `natarajan_jsu' (1006) with group `natarajan_jsu' ...
Creating home directory `/home/natarajan_jsu' ...
Copying files from `/etc/skel' ...
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
Changing the user information for natarajan_jsu
Enter the new value, or press ENTER for the default
    Full Name []: Natarajan M
    Room Number []:
    Work Phone []:
    Home Phone []:
    Other []:
Is the information correct? [Y/n] y
vmplanet@kubuntu-vm:~$
```

Figure 9: Creation of the two user accounts nmeghanathan and natarajan\_jsu

2. Run the command “sudo visudo” to launch the file “/etc/sudoers.tmp” in a text editor and insert the two lines as shown in Figure 10 (note that you will have to appropriately change the usernames depending on what you have created). Then, use Ctrl+O to save the file and press Ctrl+X to exit the editor. This step will make the two user accounts to be able to use “sudo”.

Note that the *sudo* (‘sudo’ stands for substitute **u**ser **do**) command allows users to run programs with the security privileges of another user, normally the root.

```

vmplanet: visudo
GNU nano 2.2.4 File: /etc/sudoers.tmp

# User privilege specification
root    ALL=(ALL) ALL

# Allow members of group sudo to execute any command
# (Note that later entries override this, so you might need to move
# it further down)
%sudo  ALL=(ALL) ALL
#
#includedir /etc/sudoers.d

# Members of the admin group may gain root privileges
%admin  ALL=(ALL) ALL
nmeghanathan ALL=(ALL) NOPASSWD:ALL
natarajan_jsu ALL=(ALL) NOPASSWD:ALL

```

Figure 10: Editing the /etc/sudoers.tmp file to let the two new users to run the sudo command

3. Login to nmeghanathan using the sudo command “sudo login nmeghanathan”. You would be probably prompted for the vmplanet password and definitely for the password corresponding to the user account (refer Figure 11)

```

[ VMplanet ] Kubuntu 10.10 VMware Player Devices
vmplanet: login
vmplanet@kubuntu-vm:~$ sudo login nmeghanathan
[sudo] password for vmplanet:
Password:
Linux kubuntu-vm 2.6.35-22-generic #33-Ubuntu SMP Sun Sep 19 20:34:50 UTC 2010 i686 GNU/Linux
Ubuntu 10.10

Welcome to Ubuntu!
 * Documentation:  https://help.ubuntu.com/

192 packages can be updated.
60 updates are security updates.

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

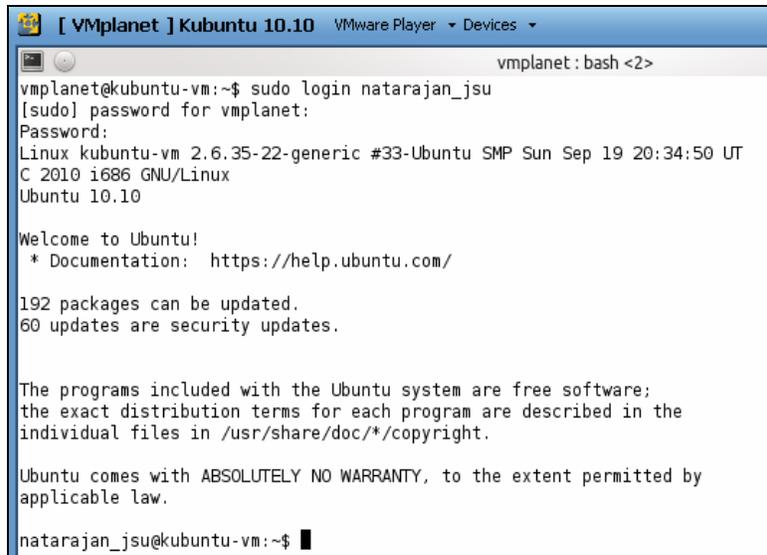
Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

nmeghanathan@kubuntu-vm:~$

```

Figure 11: Logging in to the user account nmeghanathan

4. Launch another Konsole terminal using the K-button and try to login to the other user account natarajan\_jsu, in my case, through the command: “sudo login natarajan\_jsu”
5. Generate the keys for the sender account: nmeghanathan using the command “gpg --gen-key”. Similarly for the receiver account: natarajan\_jsu. Choose the values for the algorithm parameters as indicated below for the sender and receiver accounts. Make sure to associate an email address, when asked, to each of the two accounts as discussed before.



```
[ VMplanet ] Kubuntu 10.10 VMware Player ▾ Devices ▾
vmplanet: bash <2>
vmplanet@kubuntu-vm:~$ sudo login natarajan_jsu
[sudo] password for vmplanet:
Password:
Linux kubuntu-vm 2.6.35-22-generic #33-Ubuntu SMP Sun Sep 19 20:34:50 UT
C 2010 i686 GNU/Linux
Ubuntu 10.10

Welcome to Ubuntu!
 * Documentation: https://help.ubuntu.com/

192 packages can be updated.
60 updates are security updates.

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

natarajan_jsu@kubuntu-vm:~$ █
```

Figure 12: Logging in to the user account *natarajan\_jsu*

**Choose the following parameters for both the accounts:**

Kind of Key: DSA and Elgamal  
DSA Key size: 2048 bits  
Keys does not expire at all

**Then, for each of the two accounts,** enter your real name and the email-address you want to associate with. Write a different comment for the two accounts.

Now, the system is creating the public and private keys for the account. In order to bring in more entropy (a measure of randomness) to the key creation process, which would eventually generate stronger keys - sure to do some busy things on the VM like opening a web browser (the 'rekonq' browser) and visiting a website or searching for your name in Google, etc.

**Passphrases:** Finally, you will also be required to choose a passphrase (it is your choice) and confirm it. Make sure to remember the passphrases that you assign to each public-private key pair as they are required to do any encryption/ decryption.

Finally, you should see some output like Figure 13 for the two user accounts, confirming the creating of the public and private (referred to as secret key in the output) keys.



```
natarajan_jsu@kubuntu-vm:~$ more natarajan_jsu-pk
-----BEGIN PGP PUBLIC KEY BLOCK-----
Version: GnuPG v1.4.10 (GNU/Linux)

mQMUBEOoQV4RCACUvdUPu2aIBX5n+V1DLF8K9oM4hRxOI6+YYyExQT+e+P5TWzY9
55NsRexpkPSSQFxoXeYkegyHy3x9s82wZH6qS8JA6SyOXgEkWZ0kuoeoU4Qjz0im
uS5zE9z+rJHH6o7xM51Dko85AExoIeUY4z0A46Prb6QxpHG0tf0HyMPjofET/mc
G6gg9aHc6p1qssFe1+KvPLjSJVTmCHI2ekqdlVL7gsexug0jw2uS7o1krSxCacjEJ
eS2826XQobrRGuh5SsVI9J8UfKiH27rMuf8RFP3xL/pwTxdHodPf+XJuVA5Rdoq
iXKh01hMhQ+TB70F3YIK/4j0x/b8n+nbx3pXAQDLkuuu7VdlyfMy/yDzeaEQVu6
WroOZJHuH+Mec7M5SwgAjgeRgv4z8HBffMaitW03uF8NbZv5z7iQhwZq70EmutN8
Q+2QsatiF4dSYqzEdr6HNFqSKcXxjppFeSIRlxmCFPvCda6lKYB4HlKmCEPqon4e
h5s1Z56YABVVZhwv0qxmIJMB2xD1WHdQQ/VN4hZw6swyfHTOf6nAgVFxm9ghdAwJ
9WezVwyG3N6QW7aLdLjQ5ctDL+WLL067jdhU0NHIzXmaXibI6ShYBNTJfSBjVeJ9
6zi6cktkHbKRwNhf/i3yA6kSefsZLRUaRxIn0ukpuOI3K7nmv5G1/Mpc8RQYq9R
P0hok3YlGtH+E52xrcuumTDCZzcEg8vMYhodX2uWv0f/Zu02fgl_CwBYPtVbq/T/Y
```

Figure 15: A portion of the ASCII formatted PGP Public Key Block of the Receiver (natarajan\_jsu-pk)

```
vmplanet : bash <2>
natarajan_jsu@kubuntu-vm:~$ sudo cp natarajan_jsu-pk /home/nmeghanathan/
natarajan_jsu@kubuntu-vm:~$ █
```

Figure 16: Command to copy the Receiver's public key to the Sender default folder

```
vmplanet : bash
nmeghanathan@kubuntu-vm:~$ ls -l
total 4
-rw-r--r-- 1 root root 2349 2011-01-08 05:28 natarajan_jsu-pk
nmeghanathan@kubuntu-vm:~$ █
```

Figure 17: Proof of existence of the Receiver's public key at the Sender's default folder

- 6. Importing the Receiver's public key to the Sender's key store:** Now that the receiver has exported its public-key file to the sender, the sender needs to import it. Run the command in the terminal of the sender nmeghanathan: "gpg --import natarajan\_jsu-pk"

```
nmeghanathan@kubuntu-vm:~$ gpg --import natarajan_jsu-pk
gpg: key 5F39CCE8: public key "Natarajan M (Natarajan Meghanathan hotmail account) <natarajan_jsu@hotmail.com>"
imported
gpg: Total number processed: 1
gpg:          imported: 1
nmeghanathan@kubuntu-vm:~$ █
```

Figure 18: Importing the Receiver (natarajan\_jsu-pk) public key to the sender (nmeghanathan) key store

### 7. Create and Encrypt a Message:

- Open pico or any text editor at the Sender (nmeghanathan) and enter the following line and save the file as secret-message.txt. Alternatively, you can also use the "cat > secret-message.txt" command and press Ctrl+D after entering the line (see Figure 19).

This file is created by Natarajan Meghanathan on January 8, 2011.

```
nmeghanathan@kubuntu-vm:~$ cat > secret-message.txt
This file is created by Natarajan Meghanathan on January 8, 2011.
nmeghanathan@kubuntu-vm:~$ █
```

Figure 19: Creation of the Text File to be sent

```
nmeghanathan@kubuntu-vm:~$ gpg --recipient natarajan_jsu@hotmail.com --armor --encrypt secret-message.txt
gpg: D8771124: There is no assurance this key belongs to the named user

pub 2048g/D8771124 2011-01-08 Natarajan M (Natarajan Meghanathan hotmail account) <natarajan_jsu@hotmail.com>
Primary key fingerprint: 702A 996F 187B 165B 4938 4831 4D8D 0AC5 5F39 CCE8
Subkey fingerprint: 753A BD53 5E9F CD89 606C 01CD 3076 8C7B D877 1124

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
nmeghanathan@kubuntu-vm:~$ ls -l
total 12
-rw-r--r-- 1 root root 2349 2011-01-08 05:28 natarajan_jsu-pk
-rw-r--r-- 1 nmeghanathan nmeghanathan 66 2011-01-08 05:47 secret-message.txt
-rw-r--r-- 1 nmeghanathan nmeghanathan 1006 2011-01-08 05:50 secret-message.txt.asc
nmeghanathan@kubuntu-vm:~$ █
```

Figure 20: Encryption of the secret message text file at the sender using the public key of the receiver

```
nmeghanathan@kubuntu-vm:~$ more secret-message.txt.asc
-----BEGIN PGP MESSAGE-----
Version: GnuPG v1.4.10 (GNU/Linux)

hQIOAzB2jHvYdxEkEAgAoXlwbFmtnPyhbHf0CcG1/NxQB61Fj5hqY9yVfiRD5gZ8
EPuoGODW6g1LtTu7lXLcyhh9Ni+GyHGArZADmp8pARZ8Q2GvT0+2L480e jwNO/MT
aMGeksgNjjNaDYiwz/fYUfGSntqFFpvFVLtw9RApw3YhuIbmFoU52xZ8eDRLT2t6
ERWY2bn+vTDPwHxqK+qYl6T3+hxkuu4kG6P8p2uoWSwgfZrv7pPzPZk3PFsJBmTo
Fc1ALDo33zPVKsXXtgJzH58aH5eViXt6GAaOnht5x+wKZlSdxx/TL+p0kSTqioIW
LGLFfqBJKcJyJyZnyzPIElhHBHiDPcBKYJh1QIppvggArp/WnzUHRZm1Fetwnaq
MSU2tZzXEJHBCa4/oiZnpPBVgRR7rfYyf6hhWFv82TkPjPchsM1XsCPQdYzbxRfb
kyMGNz28SGq7lzN2DNBZL5do8FIlKRrcBmpYJZsBgXEgg+eBWiuypicJwBjkl1v
JH10QYSS/PSC1mLjm9KhiPxzJtMpn6F1Rj2xrQCFRoTwbkgt7EpMEX3Lg2jf6rHH
+/KpNDbrhuV7ySxoRFflVksWfjJsADkiIzTTXwIoWeQQFWUgBSZ+n62JYVdEnJ4k
0TBE6FkQuvAMYcVFm1jb65XQY/kvKV4TqDt14RFRBhfUVt9y0gJ7VSe2/R7id49/
r9KLAa51aMoOnaMQbqJ01icmDjJNOvd153BCFmEn0FyJg878thwD9ARQ2ZSTM9Ts
eJ4xi4d/GGRNV8wRJeFgrmIbojmJMUHFIRAW/PeJpHoWDDFMVu/zG9okmKP72Jwg
ITvwt5ibFWqRaOmLkv7lP2dtGpodEDNfmiMgJAHuBaI7mfpPVMh+RyGvhPDxUw==
=I6h2
-----END PGP MESSAGE-----
nmeghanathan@kubuntu-vm:~$ █
```

Figure 21: Encrypted version of the secret message text file in ASCII format

```
[ VMplanet ] Kubuntu 10.10 VMware Player ▾ Devices ▾
nmeghanathan@kubuntu-vm:~$ sudo cp secret-message.txt.asc /home/natarajan_jsu
nmeghanathan@kubuntu-vm:~$ █
```

Figure 22: Copying the encrypted version of the secret message text file to the receiver's account

- Now, run the command (see Figure 20): “`gpg --recipient natarajan_jsu@hotmail.com --armor --encrypt secret-message.txt`” to encrypt the secret message text file using the public key of the user associated with the email address `natarajan_jsu@hotmail.com` (i.e., the receiver) and note that here we want to transform the encrypted version into an ASCII format. Press Y for accepting the public key of the receiver as stored in the key store. An encrypted version of the text file with name, `secret-message.txt.asc` would have been created. Use the `ls -l` command to see the presence of the encrypted file and run “`more secret-message.txt.asc`” to see the contents of the file (see Figure 21).
- Use the command: “`sudo cp secret-message.txt.asc /home/natarajan_jsu`” to transfer the encrypted message file to the default folder of the `natarajan_jsu` account (see Figure 22). Use the `ls -l` command at the receiver (`natarajan_jsu`) account and check the existence of the encrypted secret message file.

```
natarajan_jsu@kubuntu-vm:~$ ls -l
total 8
-rw-r--r-- 1 natarajan_jsu natarajan_jsu 2349 2011-01-08 05:24 natarajan_jsu-pk
-rw-r--r-- 1 root          root          1006 2011-01-08 05:59 secret-message.txt.asc
natarajan_jsu@kubuntu-vm:~$ █
```

Figure 23: Verification of the presence of the encrypted file at the default folder of the receiver’s account

### 8. To Decrypt the Message:

- Run the command: “`gpg --output extracted-secret-msg.txt --armor -d secret-message.txt.asc`” to decrypt the contents of the `secret-message.txt.asc` encrypted file to the named output file `extracted-secret-msg.txt`. It could be any named file, but preferably with a `.txt` extension (as we know it is indeed a text file!!) so that we can easily open the file and see its contents, as also shown next by running the `cat` command (see Figure 24 for all of these activities).

```
vmplanet: bash <2>
natarajan_jsu@kubuntu-vm:~$ gpg --output extracted-secret-msg.txt --armor -d secret-message.txt.asc
You need a passphrase to unlock the secret key for
user: "Natarajan M (Natarajan Meghanathan hotmail account) <natarajan_jsu@hotmail.com>"
2048-bit ELG-E key, ID D8771124, created 2011-01-08 (main key ID 5F39CCE8)

gpg: gpg-agent is not available in this session
gpg: encrypted with 2048-bit ELG-E key, ID D8771124, created 2011-01-08
"Natarajan M (Natarajan Meghanathan hotmail account) <natarajan_jsu@hotmail.com>"
natarajan_jsu@kubuntu-vm:~$ cat extracted-secret-msg.txt
This file is created by Natarajan Meghanathan on January 8, 2011.
natarajan_jsu@kubuntu-vm:~$ █
```

Figure 24: Decrypting the encrypted file at the receiver and viewing the contents of the extracted text file