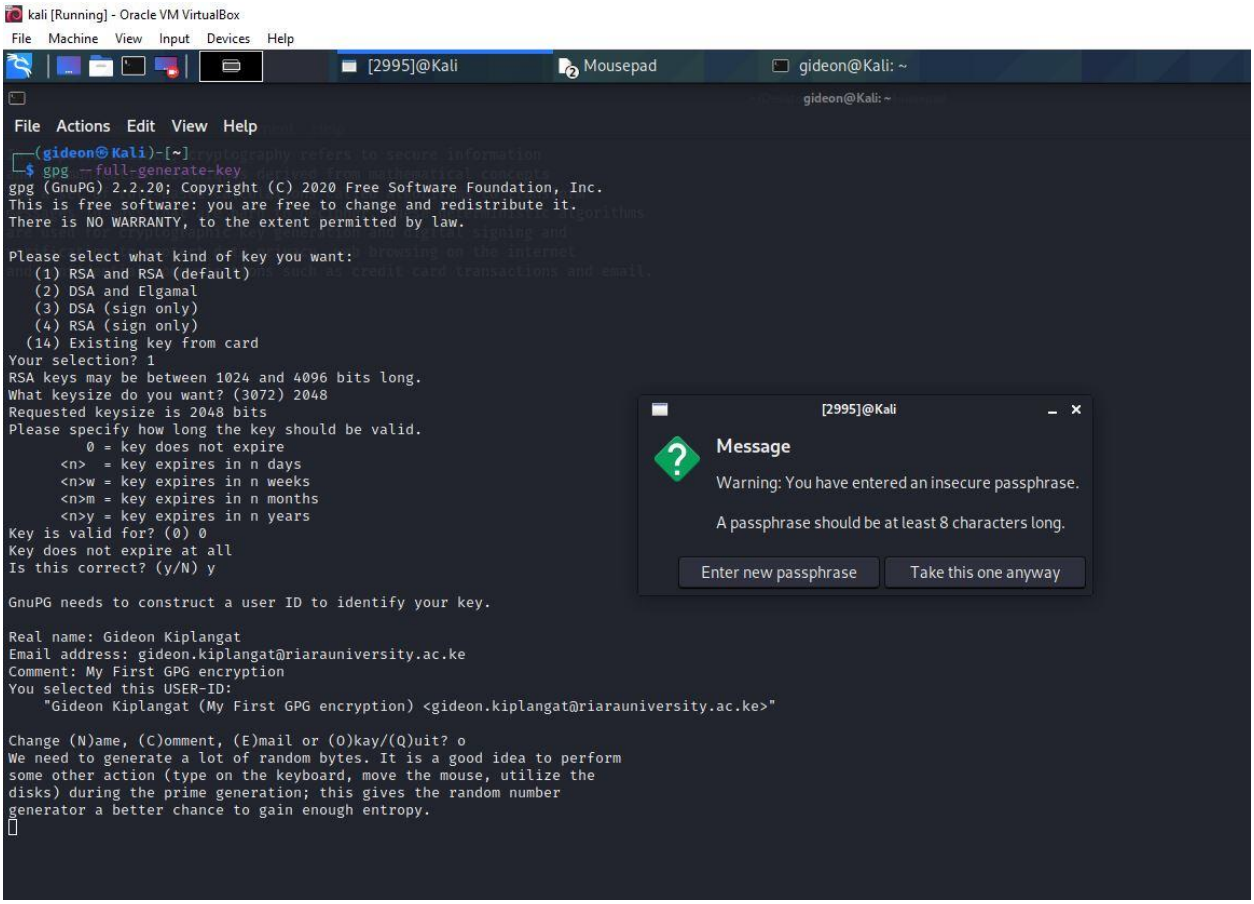
Gideon Kiplangat

Admission: 18YAD103619

RCS 402 Cryptography and information Security

Task 1: Generates Keys

- Generate keys using RSA and RSA keys option
- Key length :2048 bits long
- The key should not have an expiry date.
- After that, select whether the selection is valid or not by choosing **y**



- Enter your identity, which will be used for records in the key. **Use real credentials for this**.
  - **Realname:** (Your real first and Last Name)
  - **Email address:** (your Riara university email address)
  - **Comment:** My first GPG encryption

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

Real name: Gideon Kiplangat
Email address: gideon.kiplangat@riarauniversity.ac.ke
Comment: My First GPG encryption
```

Show screenshot

- This will be used to generate user id for the key, Select O to generate key .
- Enter the **passphrase** for your key. This will be the alias for your private key. You have created key of **2048 bits** long and since its not easy to remember, using this passphrase will enable or give you **access to your private key**.
- Choose a passphrase that is strong and easy to remember. Ensure you enter correct passphrase both times. **Don't forget this passphrase else you won't be able to access your key.**
- After entering the passphrase for your key, allow the GPG engine to generate your key. This needs generation of a lot of random bytes and may take time so its recommended **you keep your OS busy by opening another terminal and typing random text or keep your mouse moving**

```
gideon@Kali: ~
File  Actions  Edit  View  Help
What keysize do you want? (3072) 2048
Requested keysize is 2048 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
Key does not expire at all
Is this correct? (y/N) y

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

Real name: Gideon Kiplangat
Email address: gideon.kiplangat@riarauniversity.ac.ke
Comment: My First GPG encryption
You selected this USER-ID:
    "Gideon Kiplangat (My First GPG encryption) <gideon.kiplangat@riarauniversity.ac.ke>"

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.
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/gideon/.gnupg/trustdb.gpg: trustdb created
gpg: key 833430138740914F marked as ultimately trusted
gpg: directory '/home/gideon/.gnupg/openpgp-revocs.d' created
gpg: revocation certificate stored as '/home/gideon/.gnupg/openpgp-revocs.d/4DADE31F83F70C583624AC62833430138740914F.rev'
public and secret key created and signed.

pub   rsa2048 2021-07-06 [SC]
      4DADE31F83F70C583624AC62833430138740914F
uid                      Gideon Kiplangat (My First GPG encryption) <gideon.kiplangat@riarauniversity.ac.ke>
sub   rsa2048 2021-07-06 [E]


┌──(gideon㉿Kali)-[~]
```

Task 2: Encrypting and Decrypting Messages:

Use the key you have generated in step 1 to encrypt and decrypt messages

- Create a text file copy paste the below text and name your file as **yourname**.txt eg. Rose.txt

In computer science, cryptography refers to secure information and communication techniques derived from mathematical concepts and a set of rule-based calculations called algorithms to transform messages in ways that are hard to decipher. These deterministic algorithms are used for cryptographic key generation and digital signing and verification to protect data privacy, web browsing on the internet and confidential communications such as credit card transactions and email.

- Encrypt the message using public key you created. Name the output file as **gpg**.

- Display the content of the .gpg file created above using using the **cat command.**

```
┌──(gideon㉿Kali)-[~]
└─$ cd Desktop

┌──(gideon㉿Kali)-[~/Desktop]
└─$ gpg -e -r "gideon" gideon.txt
File 'gideon.txt.gpg' exists. Overwrite? (y/N) y
```

```
┌──(gideon㉿Kali)-[~/Desktop]
└─$ cat gideon.txt.gpg
```

- Decrypt the message using your public key.
- In order to decrypt this message, you need to access the private key using a passphrase you chose during key generation.
- Display the output of the decrypted file using **cat command**

```
┌──(gideon㉿Kali)-[~/Desktop]
└─$ gpg -d gideon.txt.gpg
gpg: encrypted with 2048-bit RSA key, ID 1E9D406DA89359E5, created 2021-07-06
      "Gideon Kiplangat (My First GPG encryption) <gideon.kiplangat@riarauniversity.ac.ke>"
In computer science, cryptography refers to secure information
and communication techniques derived from mathematical concepts
and a set of rule-based calculations called algorithms to transform
messages in ways that are hard to decipher. These deterministic algorithms
are used for cryptographic key generation and digital signing and
verification to protect data privacy, web browsing on the internet
and confidential communications such as credit card transactions and email.
```

```
┌──(gideon㉿Kali)-[~/Desktop]
└─$ cat gideon.txt
In computer science, cryptography refers to
secure information and communication techniques
derived from mathematical concepts and a set of
rule-based calculations called algorithms to t
ransform messages in ways that are hard to decipher.
These deterministic algorithms are used for cryptographic
key generation and digital signing and verification
to protect data privacy, web browsing on the
internet and confidential communications such
as credit card transactions and email.
```

Task 3: Signing With Private Keys

- This part takes you to the concept of digital signatures.
- The signature is provided using user's private key (and a **passphrase you entered** is required).
- Sign the textfile you created e Rose.txt using Private key and generate an output file names **yourname.sig**
- Use cat command to show the content of the file

```
┌──(gideon㉿Kali)-[~/Desktop]
└─$ gpg --sign gideon.txt
File 'gideon.txt.gpg' exists. Overwrite? (y/N) n
Enter new filename: gideon.sig
```

```
┌──(gideon㉿Kali)-[~/Desktop]
└─$ cat gideon.sig
-----BEGIN PGP SIGNED MESSAGE-----
Hash: SHA512

In computer science, cryptography refers to
secure information and communication techniques
derived from mathematical concepts and a set of
rule-based calculations called algorithms to t
ransform messages in ways that are hard to decipher.
These deterministic algorithms are used for cryptographic
key generation and digital signing and verification
to protect data privacy, web browsing on the
internet and confidential communications such
as credit card transactions and email.
-----BEGIN PGP SIGNATURE-----

iQEzBAEBCgAdFiEETa3jH4P3DFg2JKxigzQwE4dAkU8FAmDnG8YACgkQgzQwE4dA
kU8ZOAf+MqUNT2UDrveHtFMAPf5dDeQp7UXUgfkk88cPjExyZkc3sDu1q8OTr6TP
QxL8CzFvVLSZVb78qC36A5aEM4Aa3Svw1j4XpXpMWry1g8FWvh72L26V/EQAWSwC
zzkSXo9m1KF+FQcF2xrdRQVszFFtirQF1R3uqtJpWXg03+6Hb+GTj315+Jf0QvoB
UgoOX7K0s61DTVj6DOkEZtGPdxQ+RtefaZ1/UBN4Bn6+AtucGvQUjmK9LNbcz0Qg
RSghS3r0ufEmwETTdywLk18zFw5WRo4M4fOnpKh3sN+3i+jV2TIM3Ie2DL3FSEHn
07pivZ35+SaXFNMMBmzfiL0LIamRog==
=oAUU
-----END PGP SIGNATURE-----
```

Task 4: Key Revocation and Revocation Certificate Generation:

Assuming the key has been compromised by at attacker, Demonstrate how the key can be revoked and show the revoked certificate using cat command

```
┌──(gideon㉿Kali)-[~]
└─$ cat gideon.asc
-----BEGIN PGP PUBLIC KEY BLOCK-----
Comment: This is a revocation certificate

iQE2BCABCgAgFiEE/lXJ66GGJWfv8biXZQWGX52NuE8FAmDhq9sCHQIACgkQZQWG
X52NuE9eywf/SITi3EeiFqOd9IwcRlk+H19t2+rZ/0I0o7yca/zaB3GRt2qdr329
mdBfvZUSP9+7qlo9MA1E+PBOSli4+2j50FmM0dO/wi38zG113vsQc2YjXcOYJQPP
mfDhrbTY3ASSSKSdva3VH68hUYpEB8eWqvt+MuUeBwSQ6iaquJ9PT/zH8zljg963
6OUv2HbH6xvMWMmWLHkaQLemyhUl+x7ygMSgRCoDxmVXDuJJagBNt8pqmZyouCXy
+rj1rHfGktkVXNTp4qJXnJeGNoOA+aDp3kf7c9TKdhrJmzZp6Mmb/uqFmKkENKHH
gvHZdO0YHItNa3jRDvlMuWFQp6KhSNN/vQ==
=NwCl
-----END PGP PUBLIC KEY BLOCK-----
```