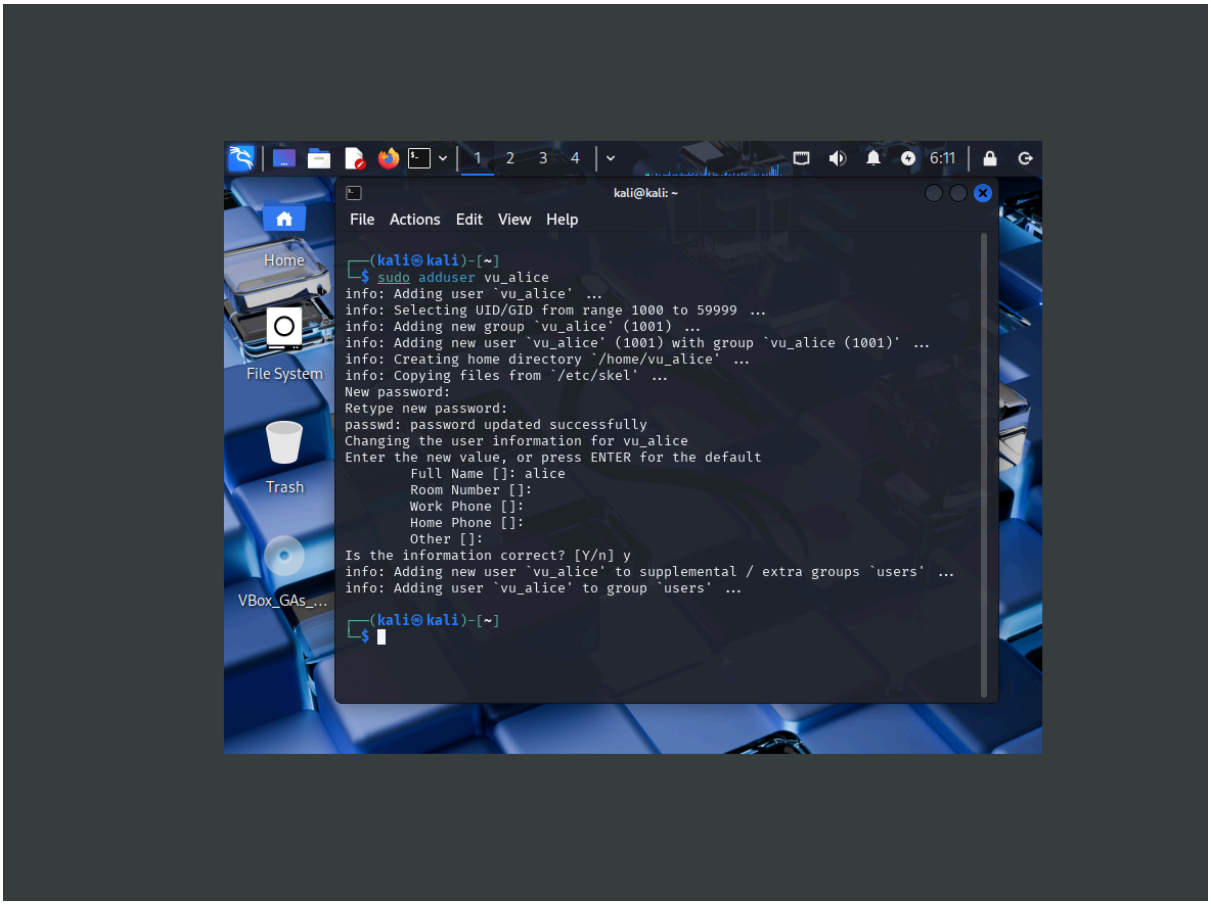
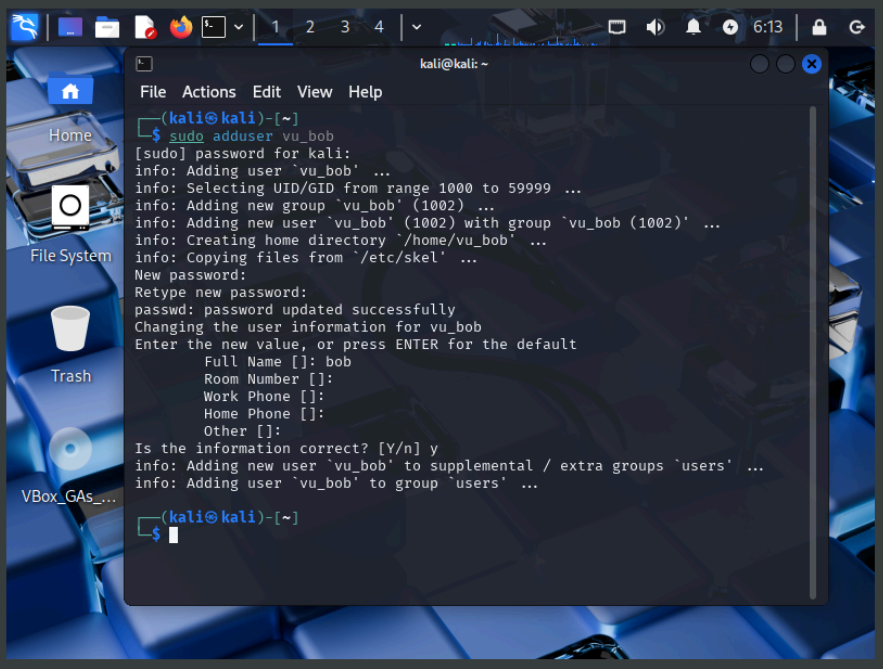
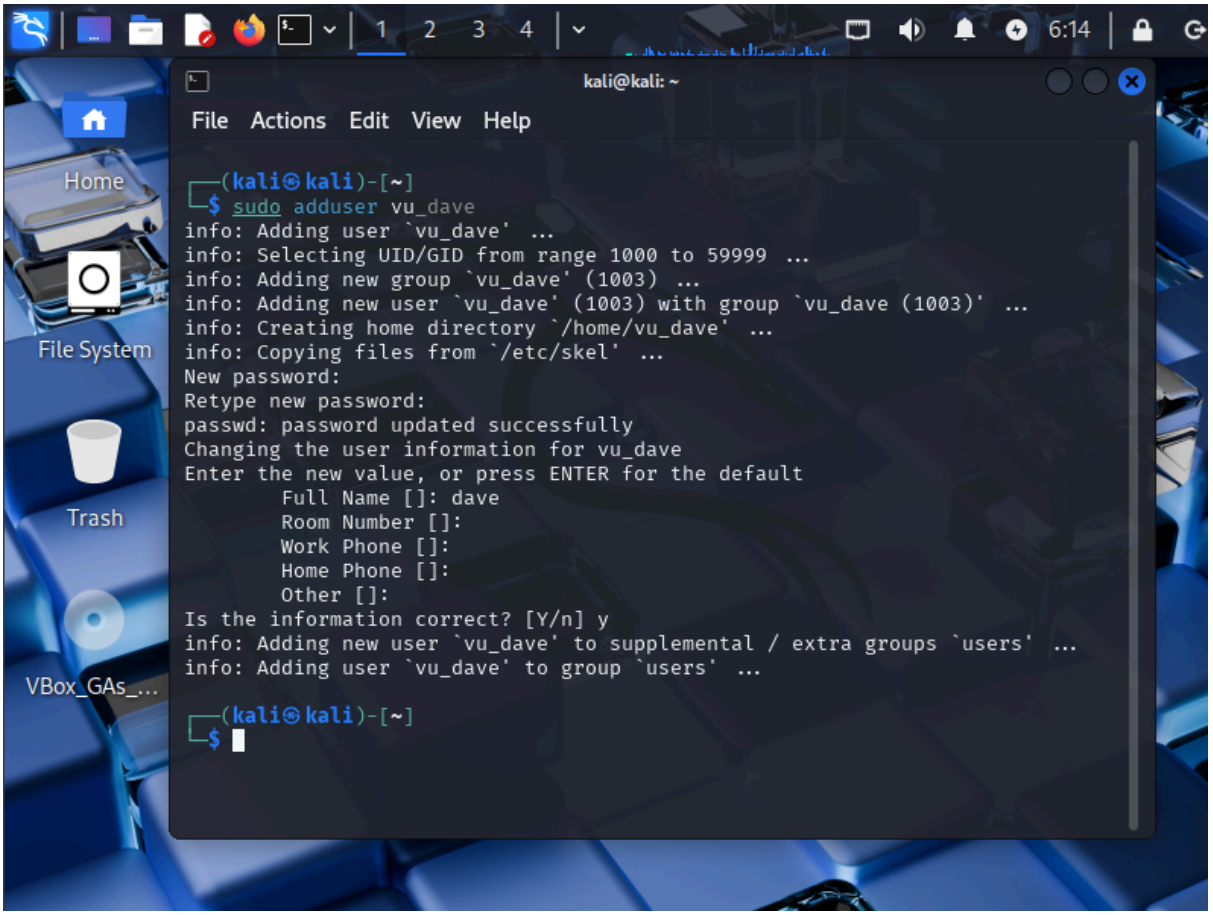
PA. Q1,a


1 2 3 4
6:11
kali@kali: ~
File Actions Edit View Help
Home
File System
Trash
VBox_GAs_...
(kali㉿kali)-[~]
$ sudo adduser vu_alice
info: Adding user `vu_alice' ...
info: Selecting UID/GID from range 1000 to 59999 ...
info: Adding new group `vu_alice' (1001) ...
info: Adding new user `vu_alice' (1001) with group `vu_alice (1001)' ...
info: Creating home directory `/home/vu_alice' ...
info: Copying files from `/etc/skel' ...
New password:
Retype new password:
passwd: password updated successfully
Changing the user information for vu_alice
Enter the new value, or press ENTER for the default
Full Name []: alice
Room Number []:
Work Phone []:
Home Phone []:
Other []:
Is the information correct? [Y/n] y
info: Adding new user `vu_alice' to supplemental / extra groups `users' ...
info: Adding user `vu_alice' to group `users' ...
(kali㉿kali)-[~]
$


1 2 3 4
6:13
kali@kali: ~
File Actions Edit View Help
Home
File System
Trash
VBox_GAs_...
(kali㉿kali)-[~]
$ sudo adduser vu_bob
[sudo] password for kali:
info: Adding user `vu_bob' ...
info: Selecting UID/GID from range 1000 to 59999 ...
info: Adding new group `vu_bob' (1002) ...
info: Adding new user `vu_bob' (1002) with group `vu_bob (1002)' ...
info: Creating home directory `/home/vu_bob' ...
info: Copying files from `/etc/skel' ...
New password:
Retype new password:
passwd: password updated successfully
Changing the user information for vu_bob
Enter the new value, or press ENTER for the default
Full Name []: bob
Room Number []:
Work Phone []:
Home Phone []:
Other []:
Is the information correct? [Y/n] y
info: Adding new user `vu_bob' to supplemental / extra groups `users' ...
info: Adding user `vu_bob' to group `users' ...
(kali㉿kali)-[~]
$

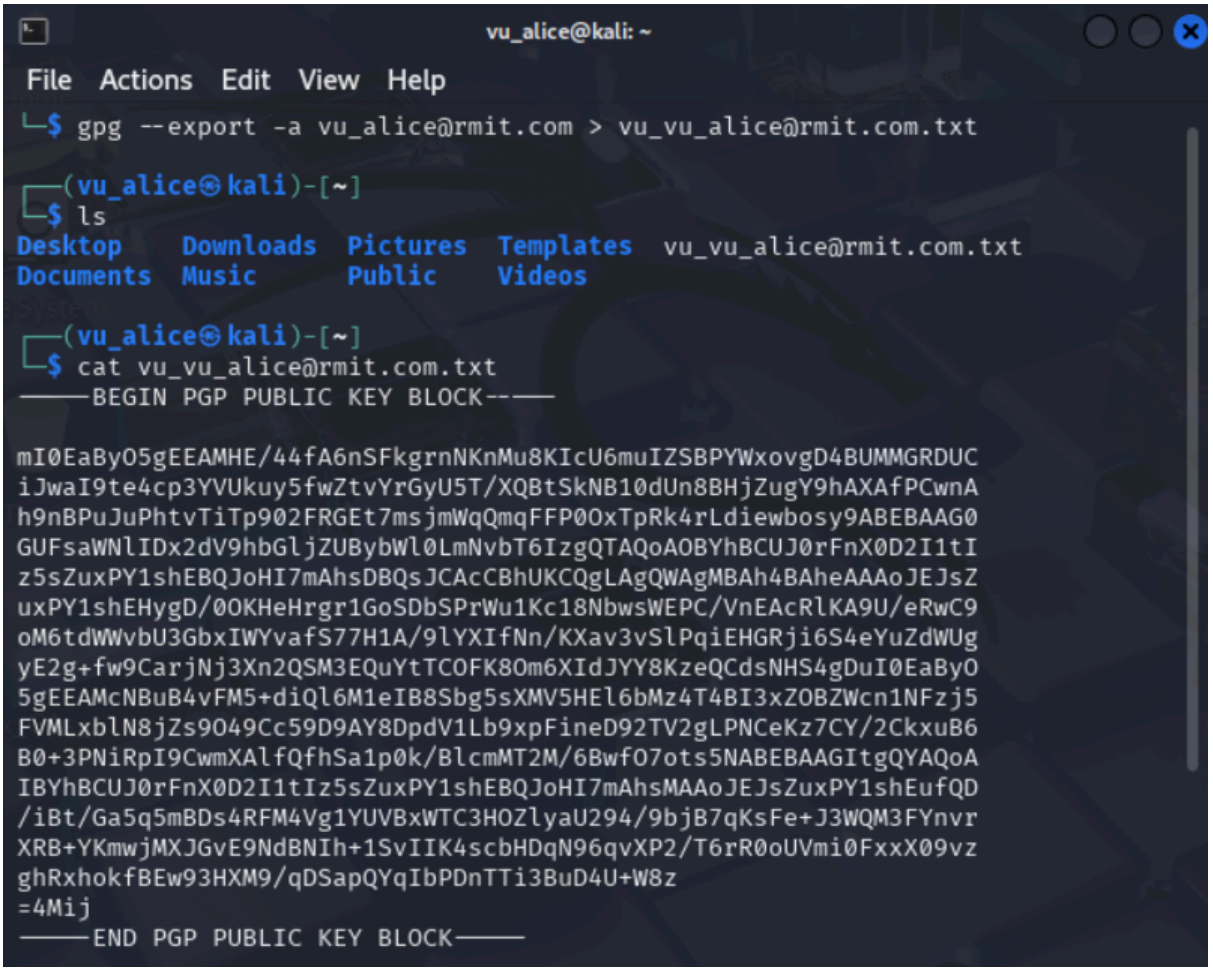


Three users, bob, dave, and alice was added using the command sudo adduser

Q1,b

```
┌──(vu_alice㉿kali)-[~]
└─$ gpg --full-generate-key
gpg (GnuPG) 2.2.46; Copyright (C) 2024 g10 Code GmbH
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)
  (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) 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: Alice
Email address: vu_alice@rmit.com
Comment:
You selected this USER-ID:
    "Alice <vu_alice@rmit.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.
We need to generate a lot of random bytes. It is a good idea to perform
```

```
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: revocation certificate stored as '/home/vu_alice/.gnupg/openpgp-revocs.d/094D308702AD896ADC5F8375BDF75DCF8DB2A864.rev'
public and secret key created and signed.

pub   rsa1024 2025-05-11 [SC]
      094D308702AD896ADC5F8375BDF75DCF8DB2A864
uid                      Alice <vu_alice@rmit.com>
sub   rsa1024 2025-05-11 [E]
```


vu_alice@kali: ~
File Actions Edit View Help
└─$ gpg --export -a vu_alice@rmit.com > vu_vu_alice@rmit.com.txt
┌──(vu_alice㉿kali)-[~]
└─$ ls
Desktop Downloads Pictures Templates vu_vu_alice@rmit.com.txt
Documents Music Public Videos
┌──(vu_alice㉿kali)-[~]
└─$ cat vu_vu_alice@rmit.com.txt
-----BEGIN PGP PUBLIC KEY BLOCK-----
mI0EaByO5gEEAMHE/44fA6nSFkgrnNKnMu8KIcU6muIZSBPYWxovgD4BUMMGRDUC
iJwaI9te4cp3YVUkuy5fwZtvYrGyU5T/XQBtSkNB10dUn8BHjZugY9hAXAfPCwnA
h9nBPuJuPhtvTiTp902FRGEt7msjmWqQmqFFP0OxTpRk4rLdiewbosy9ABEBAAG0
GUFsaWNlIDx2dV9hbGljZUBybWl0LmNvbT6IzgQTAQoAOBYhBCUJ0rFnX0D2I1tI
z5sZuxPY1shEBQJoHI7mAhsDBQsJCAcCBhUKCQgLAgQWAgMBAh4BAheAAAoJEJsZ
uxPY1shEHygD/0OKHeHrgr1GoSDbSPrWu1Kc18NbwsWEPC/VnEAcRlKA9U/eRwC9
oM6tdWWvbU3GbxIWYvafS77H1A/9lYXIfNn/KXav3vSlPqiEHGRji6S4eYuZdWUg
yE2g+fw9CarjNj3Xn2QSM3EQuYtTCOFK8Om6XIdJYY8KzeQCdsNHS4gDuI0EaByO
5gEEAMcNBuB4vFM5+diQl6M1eIB8Sbg5sXMV5HEl6bMz4T4BI3xZOBZWcn1NFzj5
FVMLxblN8jZs9O49Cc59D9AY8DpdV1Lb9xpFineD92TV2gLPNCeKz7CY/2CkxuB6
B0+3PNiRpI9CwmXAlfQfhSa1p0k/BlcmMT2M/6BwfO7ots5NABEBAAGItgQYAQoA
IBYhBCUJ0rFnX0D2I1tIz5sZuxPY1shEBQJoHI7mAhsMAAoJEJsZuxPY1shEufQD
/iBt/Ga5q5mBDs4RFM4Vg1YUVBxWTC3HOZlyaU294/9bjB7qKsFe+J3WQM3FYnvr
XRB+YKmwjMXJGvE9NdBNIh+1SvIIK4scbHDqN96qvXP2/T6rR0oUVmi0FxxX09vz
ghRxhokfBEw93HXM9/qDSapQYqIbPDnTTi3BuD4U+W8z
=4Mij
-----END PGP PUBLIC KEY BLOCK-----

```
┌──(vu_bob㉿kali)-[~]
└─$ gpg --full-generate-key
gpg (GnuPG) 2.2.46; Copyright (C) 2024 g10 Code GmbH
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)
  (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) 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: bob
Name must be at least 5 characters long
Real name: vu_bob
Email address: vu_bob@rmit.com
Comment:
You selected this USER-ID:
    "vu_bob <vu_bob@rmit.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
```

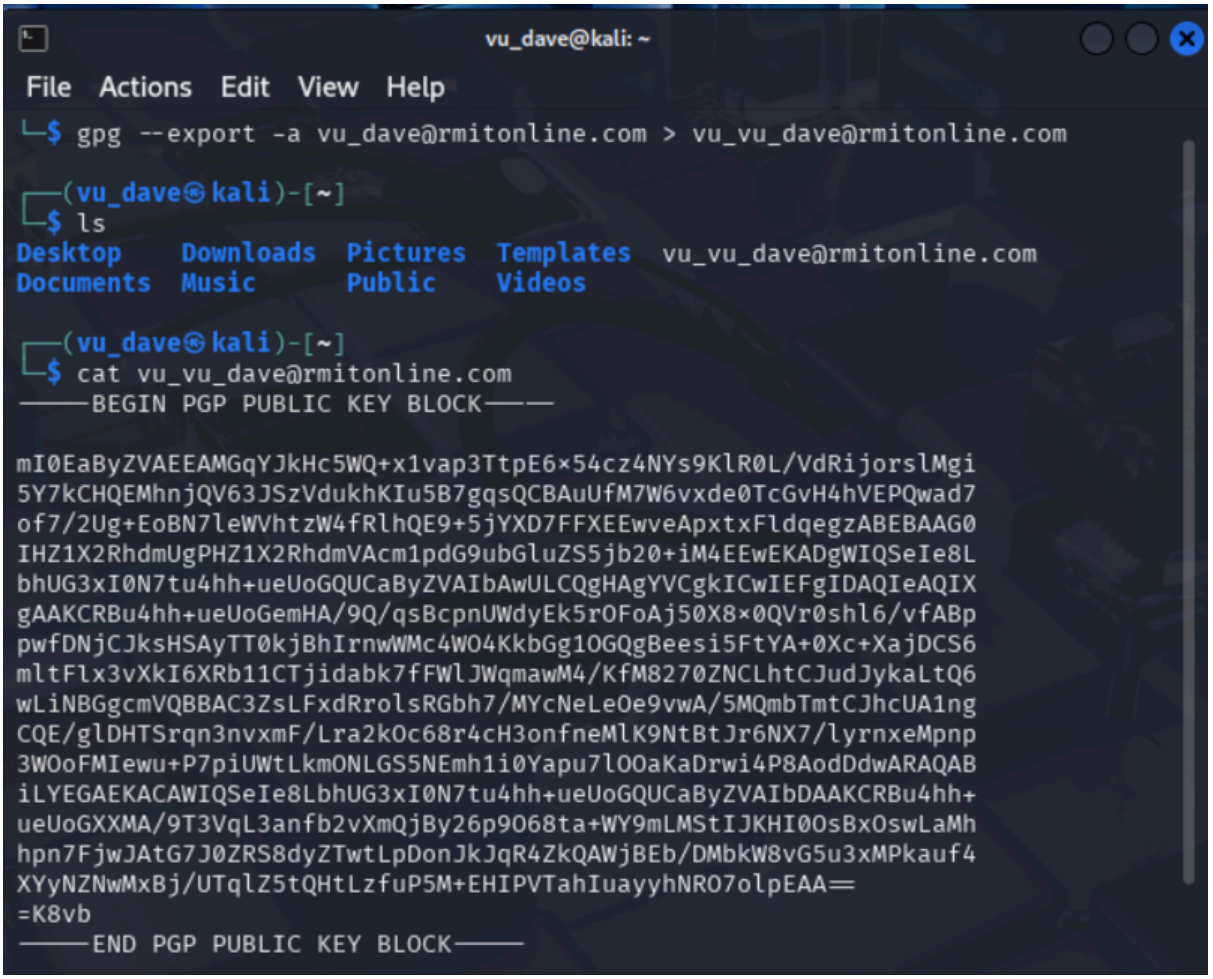
```
Change (N)ame, (C)omment, (E)mail or (O)kay/(Q)uit? o
We need to generate a lot of random bytes. It is a good idea to perform
some other action (type on the keyboard, move the mouse, utilize the
disks) during the prime generation; this gives the random number
generator a better chance to gain enough 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: revocation certificate stored as '/home/vu_bob/.gnupg/openpgp-revocs.d/03693716F0D8BC7E2F057373F64A942D3FA4358D.rev'
public and secret key created and signed.

pub   rsa1024 2025-05-11 [SC]
      03693716F0D8BC7E2F057373F64A942D3FA4358D
uid                      vu_bob <vu_bob@rmit.com>
sub   rsa1024 2025-05-11 [E]
```

```
└─$ gpg --export -armor vu_bob@rmit.com > vu_vu_bob@rmit.com.txt

┌──(vu_bob㉿kali)-[~]
└─$ ls
Desktop    Downloads  Pictures  Templates  vu_vu_bob@rmit.com.txt
Documents  Music      Public    Videos

┌──(vu_bob㉿kali)-[~]
└─$ cat vu_vu_bob@rmit.com.txt
-----BEGIN PGP PUBLIC KEY BLOCK-----

mI0EaByVtQEEAKdPZhxnrexTp564aujfCixWF4+8tnj0Ac7d9pkIxnaGdVpSMYit
umi9MTWyZaGbaYoc3FmH0w3FywbTIVsKlcXAxHGc7BKxUsenhFjth4pxRURCq/qa
NT6igBROe2pEAFE1rWTTJgt8JOyCyw8Xjq2X8ZQSyaJ8DdxU17vhN9otABEBAAG0
GFZ1IEJvYiA8dnVfYm9iQHJtaXQuY29tPojOBBMBCgA4FiEEaCE0FMh6kKIs0ST2
G+wWk0yWo9EFAmgclbUCGwMFCwkIBwIGFQoJCAsCBBYCAwECHgECF4AACgkQG+wW
k0yWo9ER7AP/RAOFcJAld9TC49RFtIy9u22QYBfeuSRHSq5mHb8szASwAiFh3UCm
LjnWxEAGF3PL9EPYpWiY+swZimN8HG/AnB+Yf31+wZ2ikVlki/CDONU/49y/NoEa
FDiq+i1rak3zWKCU2iLVDQXrx+AuSJG4BUjWyuLIN8z2Q/kZW4yxktq4jQRoHJW1
AQQAqZ+UObHE8nacrhBBeOQk54nTkAxEHOc9VnJmDBaf9WdMU2L/HdBCORhGDfDZ
HB2cU5n0/RAVYoEHBd+Lgibeuc6QwHxZzdSmtmX8XCipBmHyTSH1y505q0JVrrLr
HvTYwL05l/cuW0imeW0shHDny0mDyPMtFJgJJTm+OGIl7lMAEQEAAYi2BBgBCgAg
FiEEaCE0FMh6kKIs0ST2G+wWk0yWo9EFAmgclbUCGwwACgkQG+wWk0yWo9EXZwP+
IblwwdCX3NgcF55Nw4KTyxHVxgzo3Birfl1jTho+asazylEFT2IgVYVgAZpVmwZC
HQVXLHfDseMjNwoQd4ghOfx+7zYP/zE0lVgSeo8EYBqkcSMYNL7MGnYKzyCqAGoq
+HWh/hKvKUwEu6Arjmpf0bSK86AI2UY4xHA6/VI3+Nc=
=nJl1
-----END PGP PUBLIC KEY BLOCK-----
```

```
public and secret key created and signed.

pub   rsa1024 2025-05-08 [SC]
      9E21EF0B6E1506DF123437BB6EE2187EB9E52819
uid                      vu_dave <vu_dave@rmitonline.com>
sub   rsa1024 2025-05-08 [E]
```

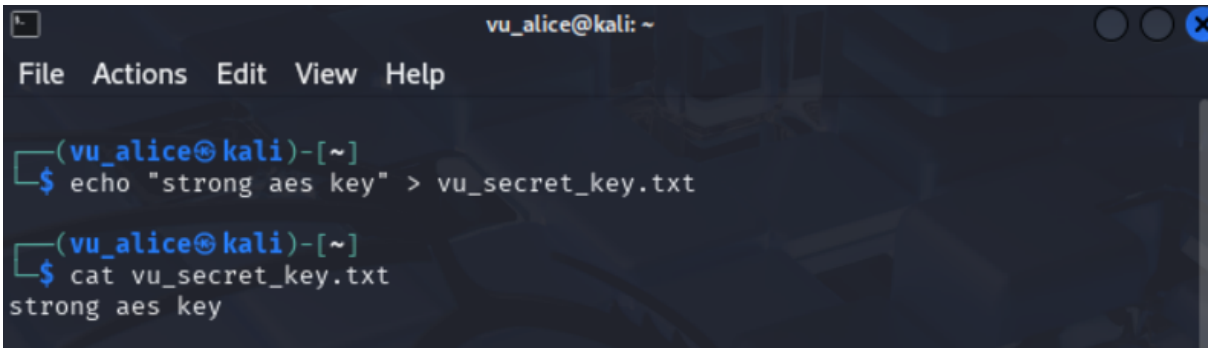
```
┌──(vu_dave㉿kali)-[~]
└─$ gpg --full-generate-key
gpg (GnuPG) 2.2.46; Copyright (C) 2024 g10 Code GmbH
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.

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)
  (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) 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: vu_dave
Email address: vu_rmit@rmitonline.com
Comment:
You selected this USER-ID:
    "vu_dave <vu_rmit@rmitonline.com>"

Change (N)ame, (C)omment, (E)mail or (O)kay/(Q)uit?
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
```

```
Change (N)ame, (C)omment, (E)mail or (O)kay/(Q)uit?
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: revocation certificate stored as '/home/vu_dave/.gnupg/openpgp-revocs.d/018ED05D5D02577F6531D3C94A7A7C83750A0EE6.rev'
public and secret key created and signed.

pub   rsa1024 2025-05-11 [SC]
      018ED05D5D02577F6531D3C94A7A7C83750A0EE6
uid                      vu_dave <vu_rmit@rmitonline.com>
sub   rsa1024 2025-05-11 [E]
```



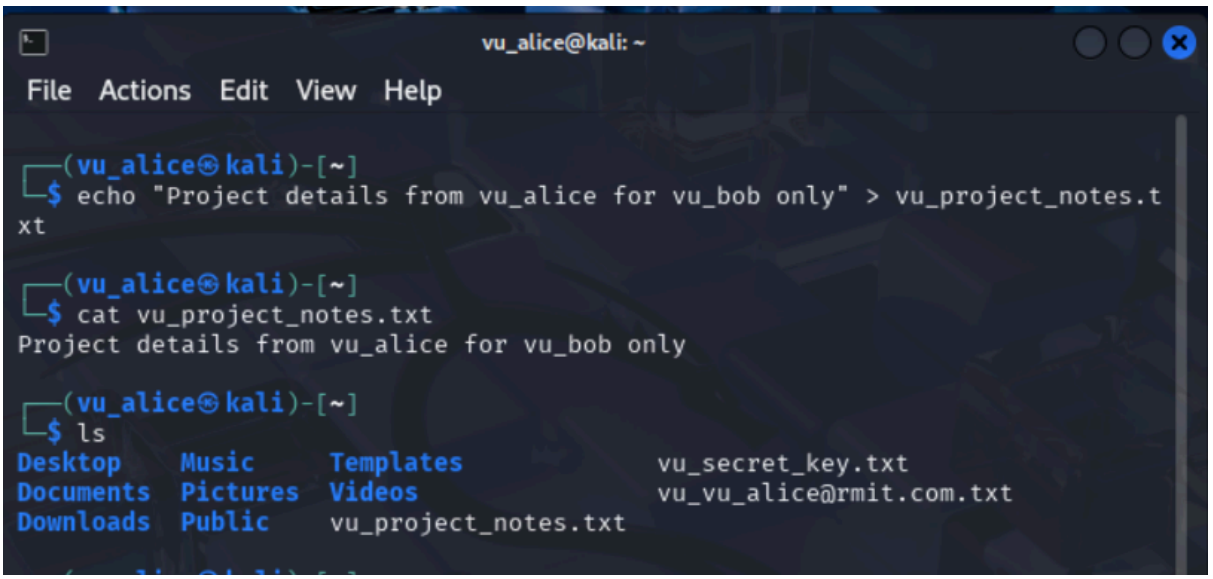
Gpg –full-generate-key was used to create an RSA key that matches the user information. This is performed on every users. Afterward gpg --export -armor was performed to extract into a readable file with –export used for exporting the public key and -armor convert it to ascii character. This is performed on Bob, dave, and alice's accounts. The command ls was performed to check whether the file was successfully created, following with cat to view the insider content of the file

Q2.a



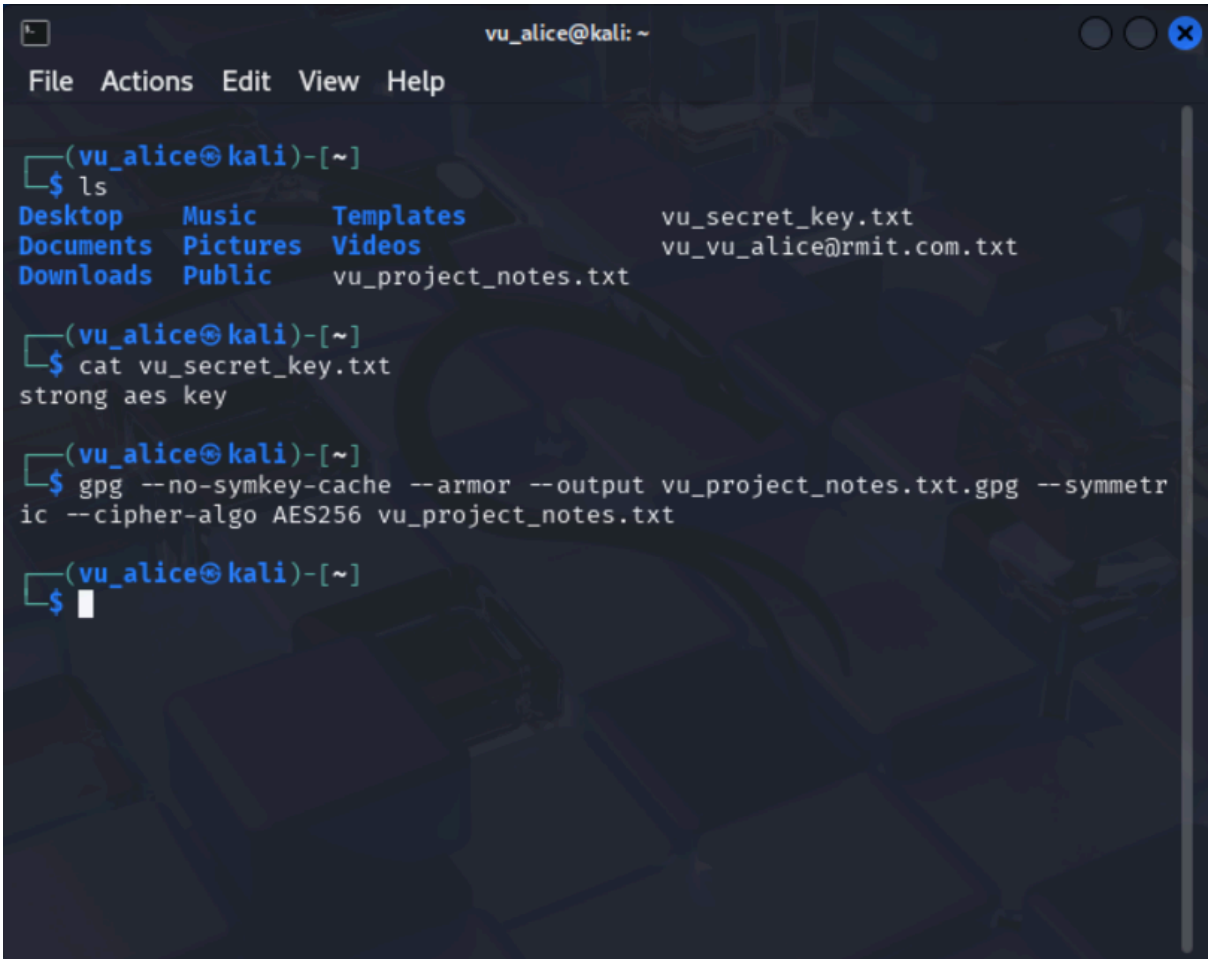
In order to performed an AES encryption, a text file was created with the key written to it using the echo command and is appended into a secret key file in alice account

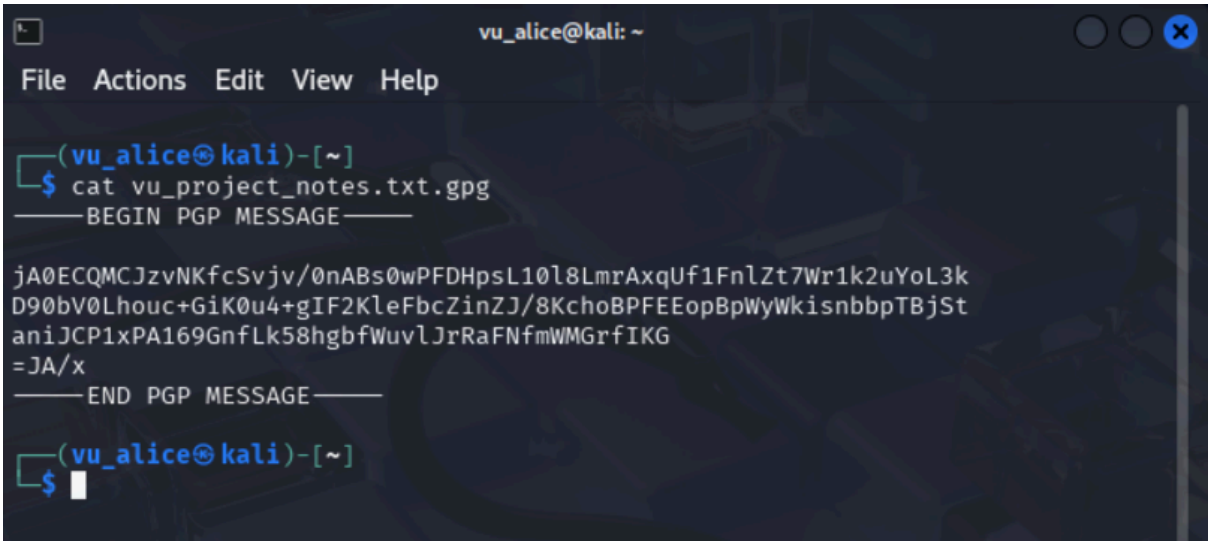
B.



Another file is created which acts as plaintext data that will be used to be encrypted. Ls and cat was used to view the content and the existence of the text file. The file will be name project notes

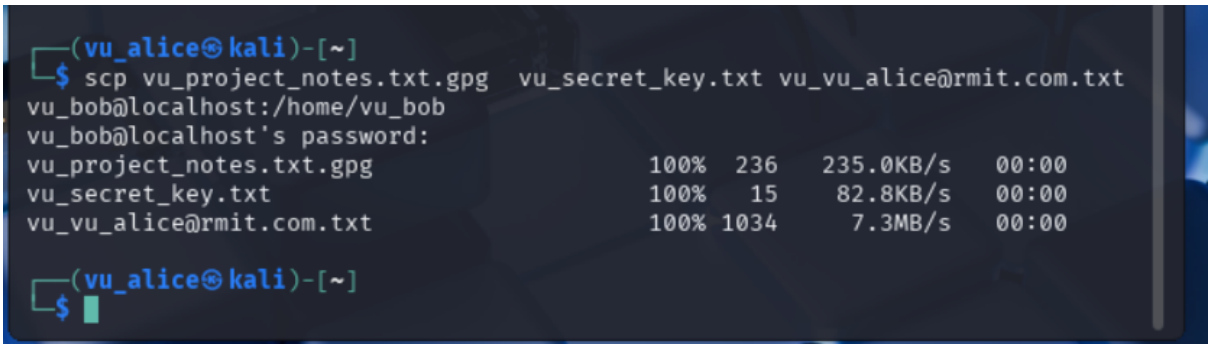
C.





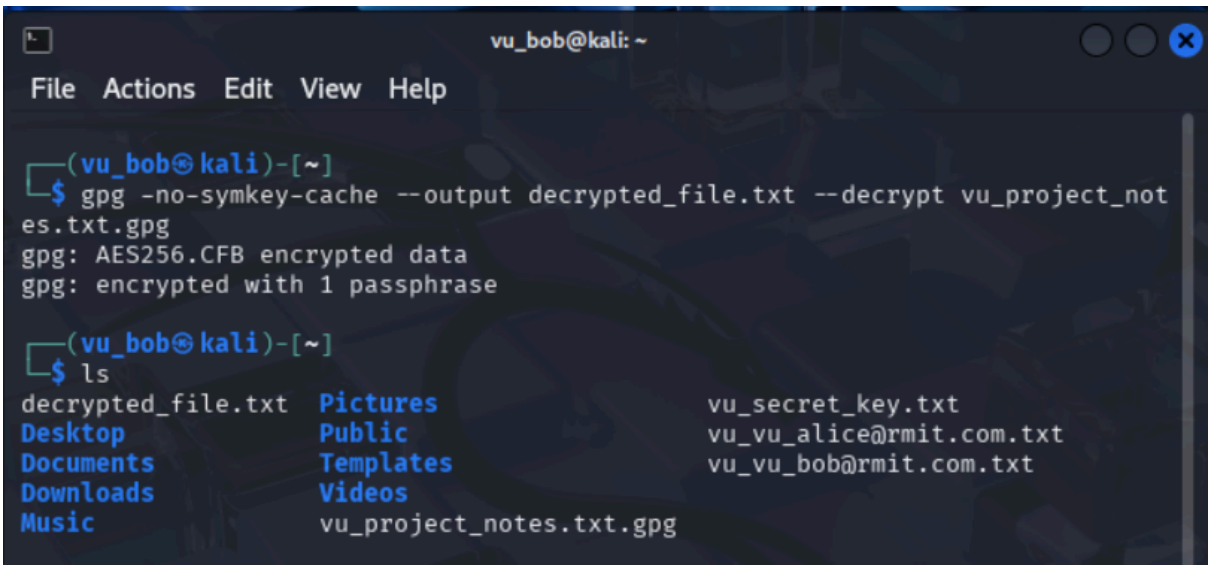
Once everything is setup, the procedure of AES encryption begin with gpg --no-symkey-cache --armor --output vu_project_notes.txt.gpg --symmetric --cipher-algo AES256 vu_project_notes.txt.
No-symkey-cache disable symmetric key cache which ensure passphrase is needed. Symmetric ensures gpg will use a symmetric encryption method. -cipher algo specifies which encryption algorithm is used. Output specifies what name the output would be. Armor ensures that it prints the output in ascii. At the end I used cat to view the content

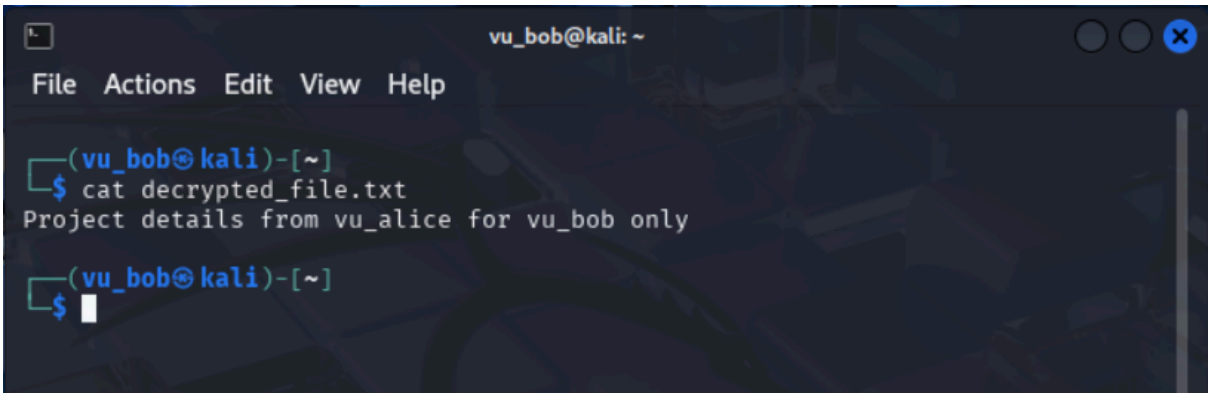
D.



Once everything was set. The three files was securely sent to bob's account through using scp. Scp ensures that the file is sent securely. The name of the files are specified and the end is the address of where scp should send those files

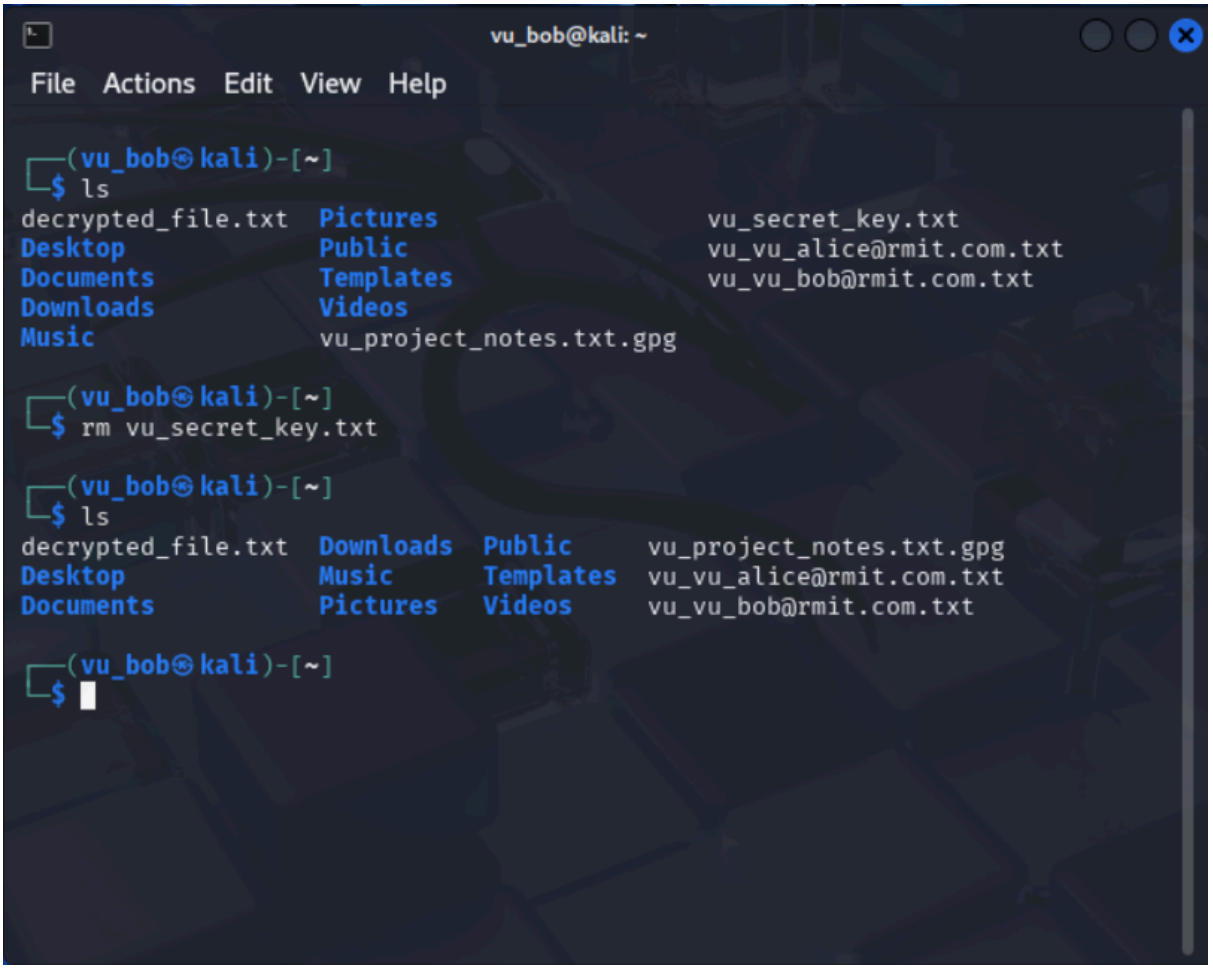
Q3,a





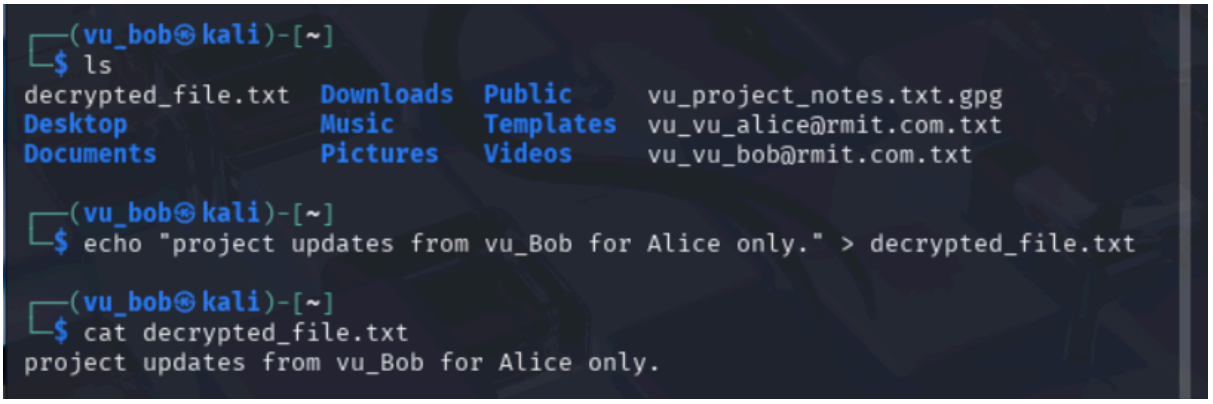
Once the packaged is successfully received by bob's account. I performed a decryption through gpg by stating what the output file should be as well as which file it is supposed to decrypt using --decrypt. No-symkey-cache ensures that the passphrase is always needed. Ls was performed confirm the existence of the file. Cat was used to view the content within the decrypted file

b



For safety precaution, the secret key was disposed using rm

c



Afterward, a new message is appended within the decrypted filed using echo

D

```
┌──(vu_bob㉿kali)-[~]
└─$ gpg --output vu_project_notes.txt.gpg --encrypt --recipient vu_alice@rmit.com decrypted_file.txt
gpg: 4554270411F9CA8C: There is no assurance this key belongs to the named user

sub  rsa1024/4554270411F9CA8C 2025-05-08 Alice <vu_alice@rmit.com>
 Primary key fingerprint: 2509 D2B1 675F 40F6 235B  48CF 9B19 BB13 D8D6 C844
      Subkey fingerprint: 040A C61E F19D 2970 B255  9D27 4554 2704 11F9 CA8C

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
File 'vu_project_notes.txt.gpg' exists. Overwrite? (y/N) y

┌──(vu_bob㉿kali)-[~]
└─$
```

Using the given public RSA key within the vu_vu_alice@rmit.com.txt file, the public key is imported using --import for usage of encryption. Gpg command with --output to specify the name --encrypt to encrypt the data using RSA encryption --recipient state which receiver is it intended for. And the specific file that I want to encrypt at the end

e

```
┌──(vu_bob㉿kali)-[~]
└─$ scp vu_project_notes.txt.gpg vu_alice@localhost:/home/vu_alice
vu_alice@localhost's password:
vu_project_notes.txt.gpg                                      100%  265   125.4KB/s   00:00
```

```
┌──(vu_alice㉿kali)-[~]
└─$ ls
Desktop  Documents  Downloads  Music  Pictures  Public  Templates  Videos  vu_project_notes.txt  vu_project_notes.txt.gpg  vu_secret_key.txt  vu_vu_alice@rmit.com.txt

┌──(vu_alice㉿kali)-[~]
└─$ cat vu_project_notes.txt
Project details from vu_alice for vu_bob only

┌──(vu_alice㉿kali)-[~]
└─$ cat vu_project_notes.txt.gpg
ss2w~◆A◆◆`◆◆≥_j`◆s◆W◆◆◆f
◆S

p2◆◆!r◆◆◆憫◆7◆◆δ◆,◆"\(κ◆◆c◆◆◆◆N ,m"◆◆◆◆◆␑,◆'◆QPc◆,Xb◆◆◆◆x◆◆◆9*X◆◆◆-8◆◆◆◆*
j◆e◆◆Qc◆VQ◆z◆◆x◆s◆o◆YaJg◆P◆◆aSq◆#▒W8◆◆◆◆8,D]6M3◆P"_-%                    B◆▒◆q◆y◆◆◆G◆◆#^_      ʰ◆E◆]◆◆z◆◆
┌──(vu_alice㉿kali)-[~]
└─$
```

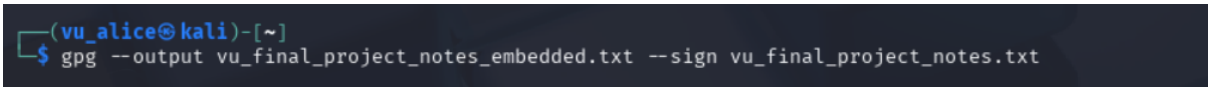
The file is then securely sent to Alice's accounts. Ls to check if the file got through and cat to view the inside of the content

Q4 a,b

```
┌──(vu_alice㉿kali)-[~]
└─$ gpg --output vu_final_project_notes.txt --decrypt vu_project_notes.txt.gpg
gpg: encrypted with 1024-bit RSA key, ID 4554270411F9CA8C, created 2025-05-08
      "Alice <vu_alice@rmit.com>"

┌──(vu_alice㉿kali)-[~]
└─$ ls
Desktop    Downloads  Pictures  Templates  vu_final_project_notes.txt  vu_project_notes.txt.gpg  vu_vu_alice@rmit.com.txt
Documents  Music      Public    Videos     vu_project_notes.txt        vu_secret_key.txt

┌──(vu_alice㉿kali)-[~]
└─$ cat vu_final_project_notes.txt
project updates from vu_Bob for Alice only.

┌──(vu_alice㉿kali)-[~]
└─$
```

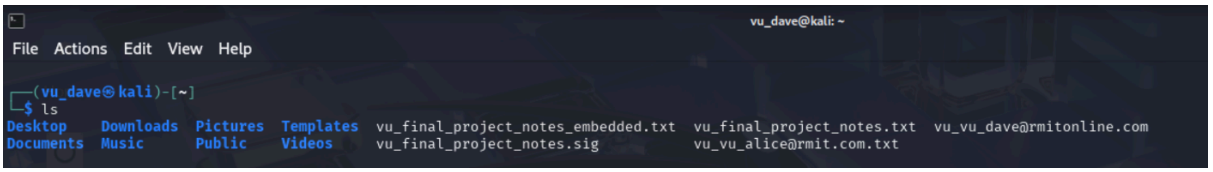
The file is then decrypted using Alice's own private key and named vu_final_projects_notes.txt. Cat is used to view the decrypted content inside

Q5,a



A gpg command is prepared followed by --detach-sign to create a detached signature output of the chosen file. Another gpg command with –sign is to create an embedded signature

B



The file is sent to dave using scp

C

The file was sent to dave. And it was verified using gpg --verify

CR

Q1

Alice should use a symmetric encryption technique. The reason being that Alice is looking for an encryption method that helps her transfer data in a secure manner, not for signing or verification. Asymmetric encryption is usually primarily used for verification and signing [1, p. 24, sec. 3.3]. Symmetric encryption such as AES encryption is much faster and lighter. Making secure data

transferring more efficient [1, pg. 20, sec. 3.2]. Alice can use command like gpg --symmetric --cipher-algo AES256 file.txt to encrypt her file.txt

Q2

Let's say Alice were to choose asymmetric encryption as her encryption method. One of the main challenges is the speed of secured data transferring. Asymmetric encryption is designed to sign and verify the ownership of a file. It is computationally intensive and takes more time to generate, in comparison to symmetric encryption [1, p. 24, sec. 3.3].

Because of it, it is ill advised to use it as a secured data transferring method. Symmetric encryption method fits what Alice needs in that, while the same key can be used to both encrypt and decrypt. If Alice is to follow the recommended security practice. The key is usually discarded immediately after being successfully transferred. Making it a quick and easy solution

DI

Q1:

The [8]CIA Triad consists of three principles, which are Confidentiality, Integrity, and Availability.

Within this context that is applied to the EHR system. Confidentiality ensures that the sensitive information and personal data is safe and can only be accessed by trusted and authorized staff members. One of the key risks that could potentially threaten the confidentiality of the EHR system would be phishing attacks. A prominent case study of this is the UVM health Network Ransomware Attack [2]. In October 2020, a worker from UVM Health clicked on a link that was sent to him through an email. It was stated that the email "seemed legitimate". Due to the accident, the data of the organization was compromised. And posted as a risk to confidential principle

Another principle is integrity. Within the context of the EHR system, this usually means the data of the patient are accurate and trustworthy. One of the key risks that could potentially threaten the integrity of these data would be ransome doing altercation to the data. [3]An example of this is in February of 2024, a cyberattack was initiated against Change Healthcare. The data was breached due to a lack in security practice and approximately six terabytes worth of information was compromised. This is posted as a risk to the integrity principle as the attackers have the power to change and alter the data. Essentially rendering the data as unreliable

The third principle of the CIA Triad is availability. Within the context, Availability essentially means that the EHR System is ready and accessible. One of the key risks that could potentially threaten the Availability of the EHR system is a DDoS attack. [4] An example of this can be seen in the Singapore Public Healthcare DDos attack in November 2023, where an unknown attacker flooded the system with requests. This slowed down the system and caused a DDoS, and the services were disrupted. Because of it, staff members weren't able to access the needed data and records and were rendered as unavailable for a brief period of time. Affectively posted as a risk to the availability principle

Q2:

With the three case studies, some security controls can be carried out to minimise the risk and preserve the three principles of the EHR System.

With the UVM data breach we learned that violation of data confidentiality can happen with phishing attacks. To ensure something like this wouldn't likely to happen with the EHR system, the implementation of Role-Based Access Control would ensure that there's a less likely chance it will happen again [6]. With RBAC implemented, it means that the accounts of users within an organisation are compartmentalized. So in the case of an account got compromised due to a phishing attack or any other attacks, the attacker would have limited access to the patients' records, preserving the CIA Confidentiality principle .

With the cyberattack against Change Healthcare in 2024, The uncertainty of the data integrity and data compromise came due to the lack of backup. This means Change Healthcare cannot ensure the data is reliable after the attack. [7] To ensure the integrity is not compromised with the EHR system, an implementation of a secured data backup and contingency plan could be placed. Having a backup ensures that the data can always be restored if a cyberattack happens.

[1] National Institute of Standards and Technology, Guideline for Using Cryptographic Standards in the Federal Government: Cryptographic Mechanisms (SP 800-175B), Mar. 2016. [Online]. Available: https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-175B.pdf

[2] K. Young, "Cyber Case Study: UVM Health Network Ransomware Attack," CoverLink Insurance | Ohio Independent Insurance Agency, Dec. 06, 2021. https://coverlink.com/case-study/uvm-health-network-ransomware-attack/

[3] INSURICA, "Cyber Case Study: Change Healthcare Cyberattack | INSURICA," INSURICA, Oct. 17, 2024. https://insurica.com/blog/cyber-case-study-change-healthcare-cyberattack/

[4] "Internet connectivity for Public Healthcare Institutions affected by Distributed Denial-of-Service (DDoS) attack," www.synapxe.sg. https://www.synapxe.sg/media-releases/corporate/internet-connectivity-for-public-healthcare-institutions-affected-by-ddos-attack

[6] NIST, "Security and Privacy Controls for Information Systems and Organizations," Security and Privacy Controls For Information Systems and Organizations, vol. 5, no. 5, Sep. 2020, doi: https://doi.org/10.6028/nist.sp.800-53r5.

[7] "PROTECTING DATA FROM RANSOMWARE AND OTHER DATA LOSS EVENTS A Guide for Managed Service Providers to Conduct, Maintain and Test Backup Files OVERVIEW." Available: https://www.nccoe.nist.gov/sites/default/files/legacy-files/msp-protecting-data-extended.pdf

[8]*Instructure.com*, 2025.
https://rmit.instructure.com/courses/144018/pages/week-1-lecture?module_item_id=7199165