

Using Encryption to Enhance Confidentiality and Integrity (4e)

Fundamentals of Information Systems Security, Fourth Edition - Lab 05

Student:

Brandon Trinkle

Email:

btrinkle52@gmail.com

Time on Task:

1 hour, 27 minutes

Progress:

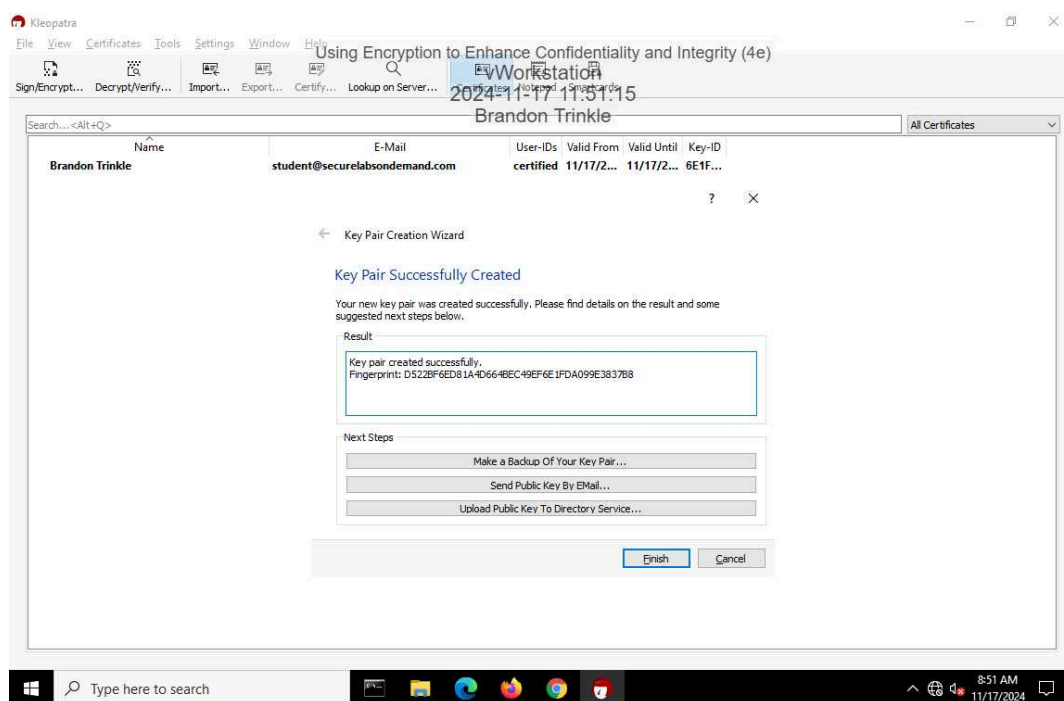
100%

Report Generated: Sunday, November 17, 2024 at 1:11 PM

Section 1: Hands-On Demonstration

Part 1: Create and Exchange Asymmetric Encryption Keys

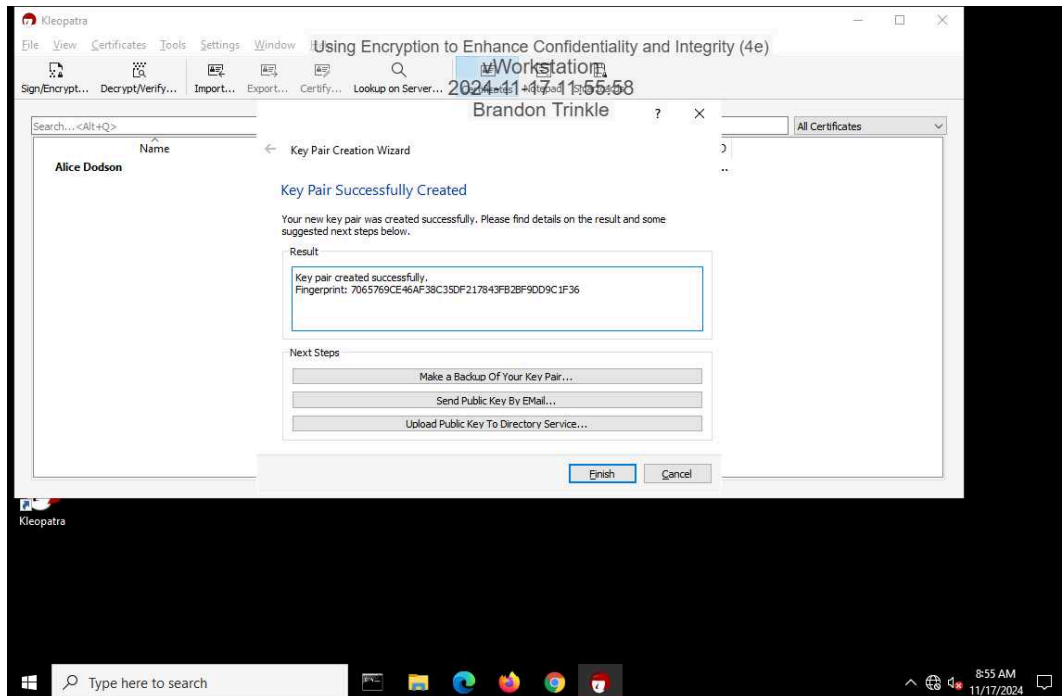
9. Make a screen capture showing the fingerprint for your key pair.



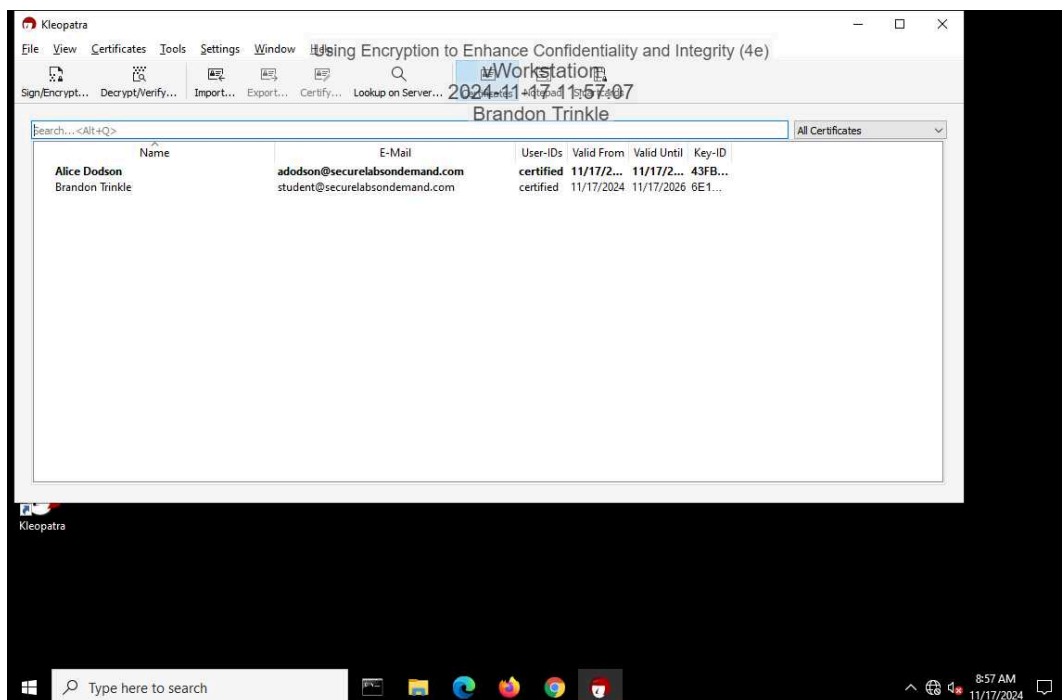
Using Encryption to Enhance Confidentiality and Integrity (4e)

Fundamentals of Information Systems Security, Fourth Edition - Lab 05

22. Make a screen capture showing the fingerprint for Alice's key pair.



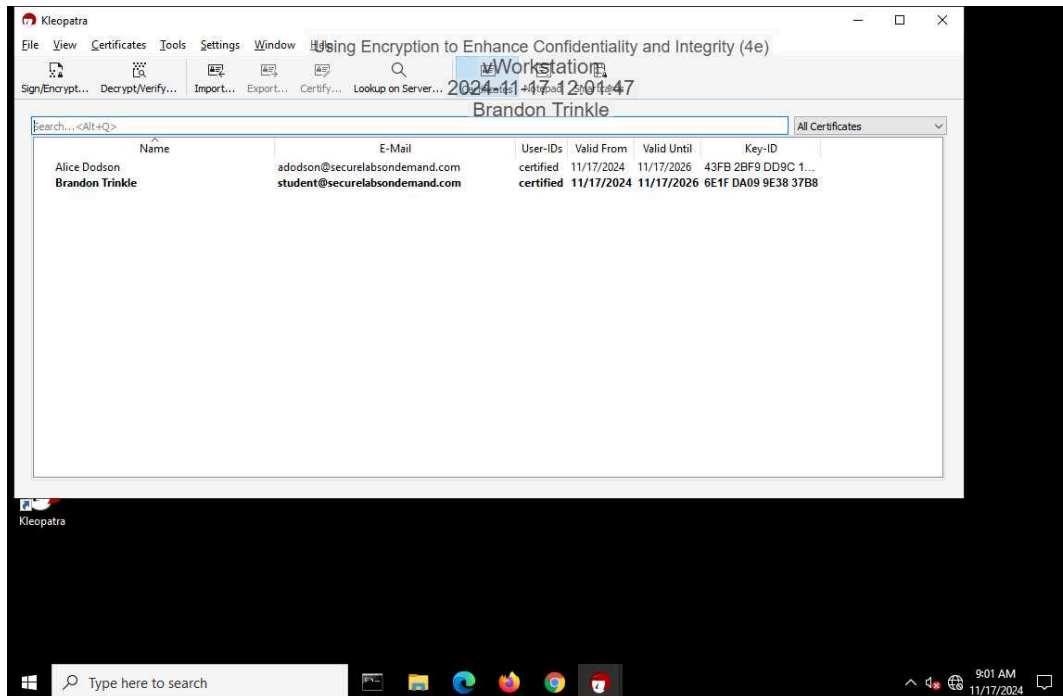
30. Make a screen capture showing your public key in Alice's certificate cache.



Using Encryption to Enhance Confidentiality and Integrity (4e)

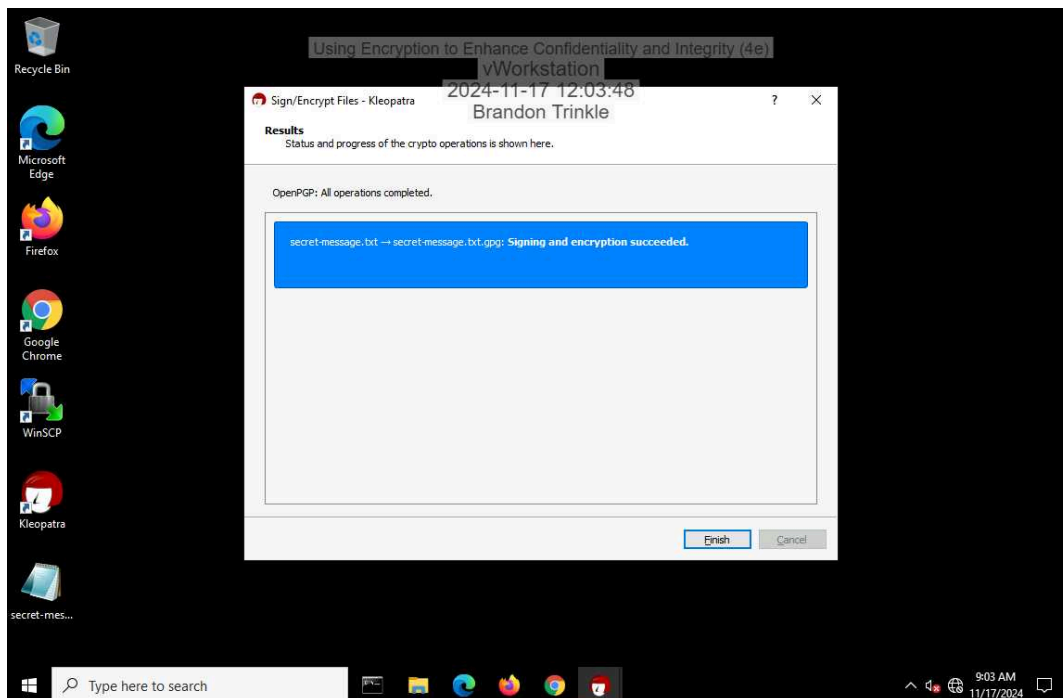
Fundamentals of Information Systems Security, Fourth Edition - Lab 05

35. Make a screen capture showing Alice's public key in your certificate cache.



Part 2: Encrypt a File Using Asymmetric Encryption

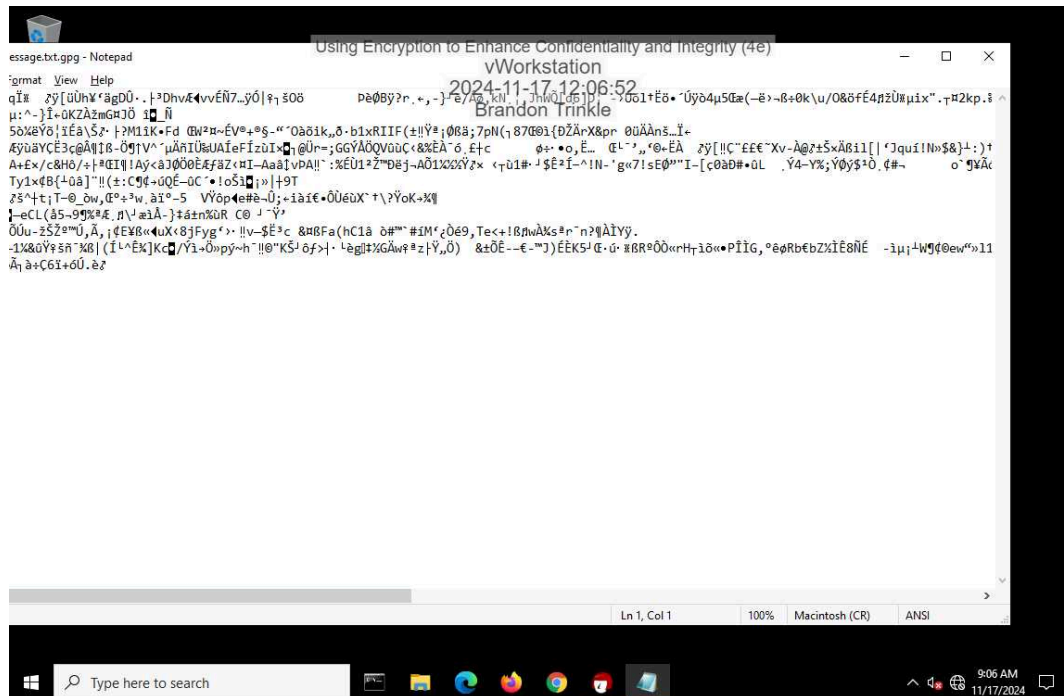
9. Make a screen capture showing the successful signing and encryption message.



Using Encryption to Enhance Confidentiality and Integrity (4e)

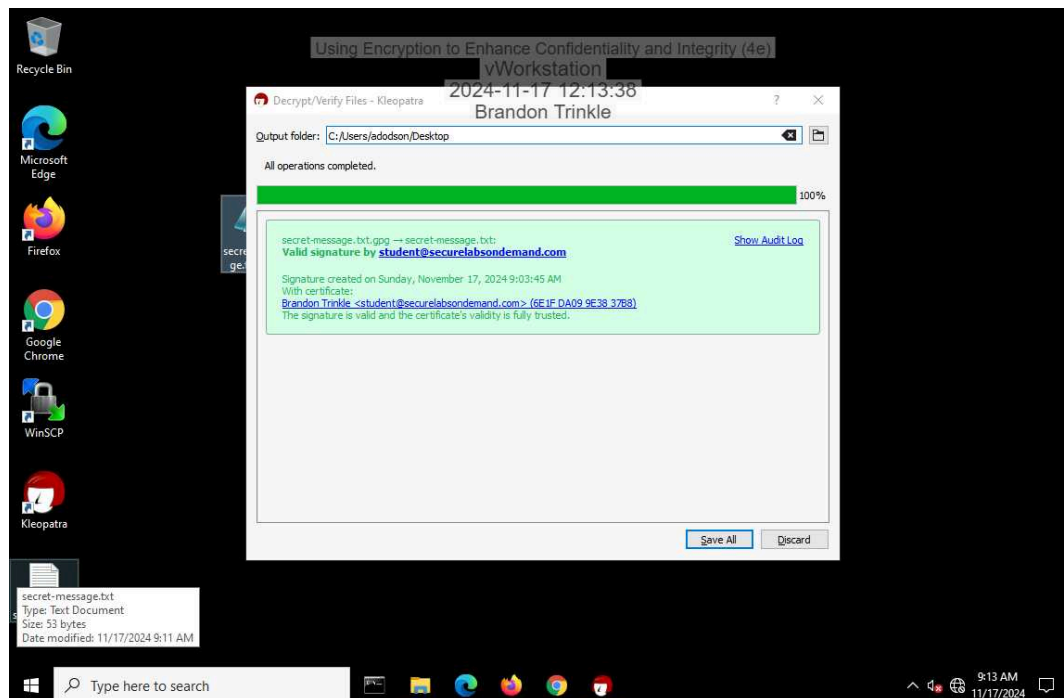
Fundamentals of Information Systems Security, Fourth Edition - Lab 05

12. Make a screen capture showing the ciphertext.



Part 3: Decrypt a File Using Asymmetric Encryption

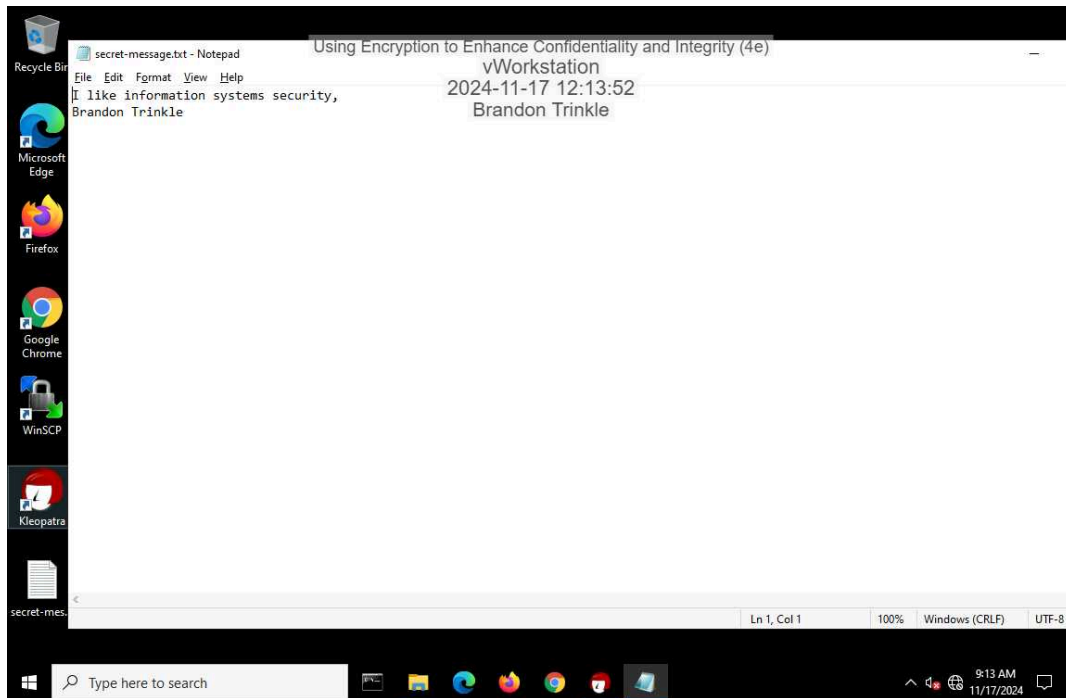
15. Make a screen capture showing the Decrypt/Verify Files window.



Using Encryption to Enhance Confidentiality and Integrity (4e)

Fundamentals of Information Systems Security, Fourth Edition - Lab 05

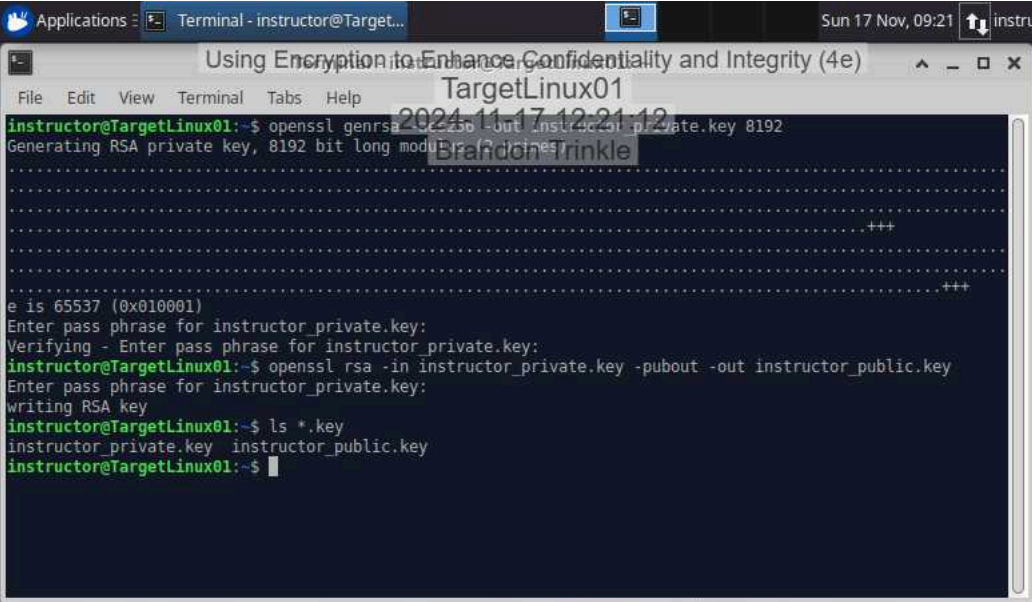
18. Make a screen capture showing the **decrypted secret-message.txt** file in Notepad.



Section 2: Applied Learning

Part 1: Create an Asymmetric Key Pair

10. Make a screen capture showing the instructor's key pair files.



The screenshot shows a terminal window titled "Terminal - instructor@Target..." with a window manager title bar that reads "Using Encryption to Enhance Confidentiality and Integrity (4e)". The terminal output is as follows:

```
instructor@TargetLinux01:~$ openssl genrsa -pubout instructor_private.key 8192
Generating RSA private key, 8192 bit long modulus...+++++
.....+++++
e is 65537 (0x010001)
Enter pass phrase for instructor_private.key:
Verifying - Enter pass phrase for instructor_private.key:
instructor@TargetLinux01:~$ openssl rsa -in instructor_private.key -pubout -out instructor_public.key
Enter pass phrase for instructor_private.key:
writing RSA key
instructor@TargetLinux01:~$ ls *.key
instructor_private.key  instructor_public.key
instructor@TargetLinux01:~$
```

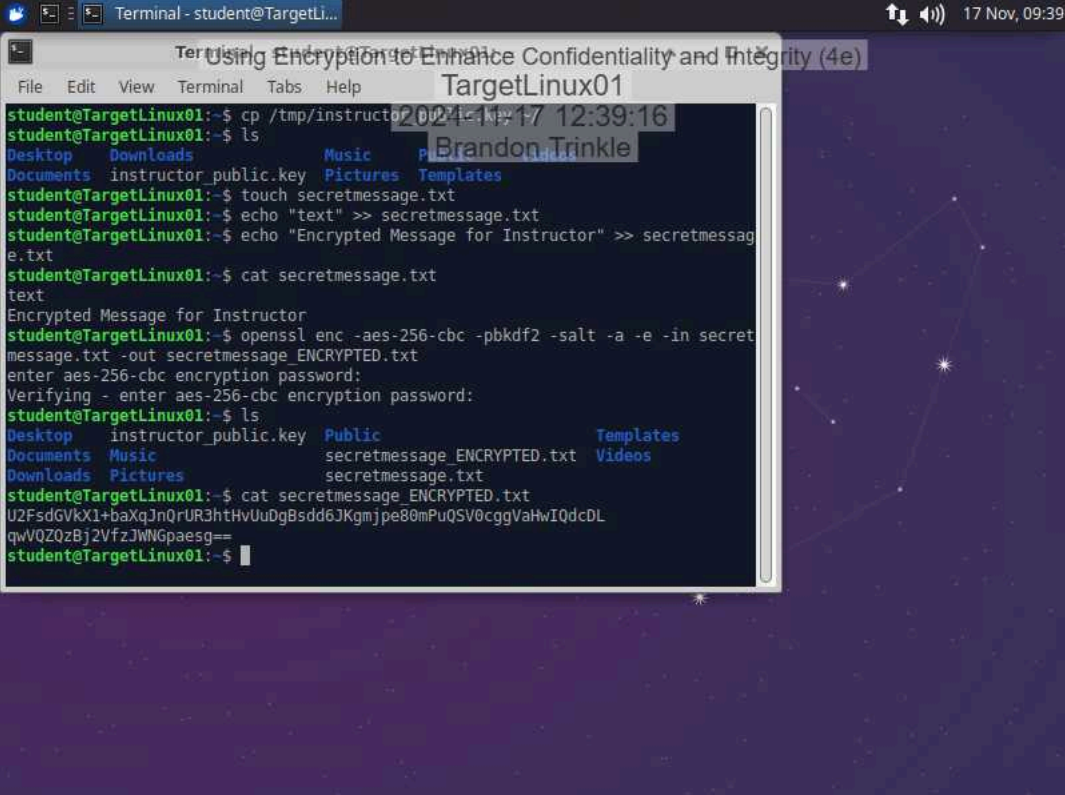
Watermarks for "TargetLinux01", "2024-11-17 12:21:12", and "Brandon Trinkle" are visible over the terminal output.

Part 2: Encrypt a File Using Symmetric Encryption

11. Document the password you used to symmetrically encrypt the file.

security

13. Make a screen capture showing the ciphertext in the `secretmessage_ENCRYPTED.txt` file.



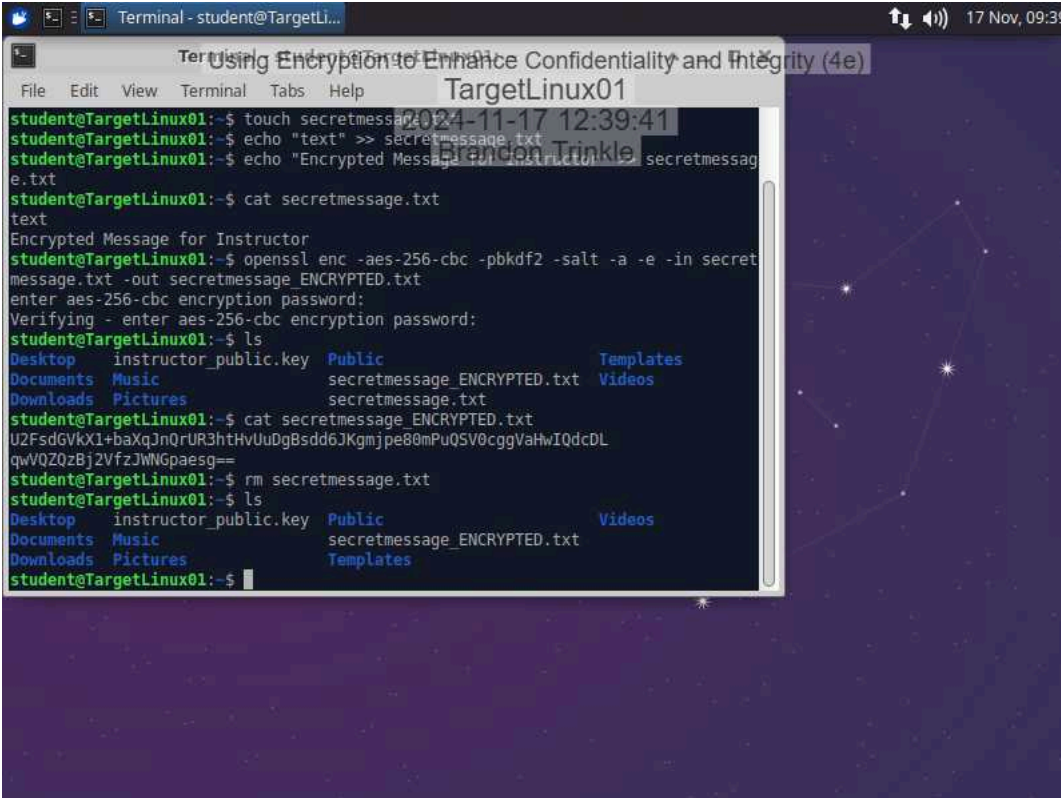
The screenshot shows a terminal window titled "Terminal - student@TargetLinux01" with a menu bar (File, Edit, View, Terminal, Tabs, Help) and a title bar (TargetLinux01). The terminal output is as follows:

```
student@TargetLinux01:~$ cp /tmp/instructor_public.key ./
student@TargetLinux01:~$ ls
Desktop  Downloads  Music  Pictures  Templates
Documents  instructor_public.key  secretmessage.txt
student@TargetLinux01:~$ touch secretmessage.txt
student@TargetLinux01:~$ echo "text" >> secretmessage.txt
student@TargetLinux01:~$ echo "Encrypted Message for Instructor" >> secretmessage.txt
student@TargetLinux01:~$ cat secretmessage.txt
text
Encrypted Message for Instructor
student@TargetLinux01:~$ openssl enc -aes-256-cbc -pbkdf2 -salt -a -e -in secretmessage.txt -out secretmessage_ENCRYPTED.txt
enter aes-256-cbc encryption password:
Verifying - enter aes-256-cbc encryption password:
student@TargetLinux01:~$ ls
Desktop  instructor_public.key  Public  Templates
Documents  Music  secretmessage_ENCRYPTED.txt  Videos
Downloads  Pictures  secretmessage.txt
student@TargetLinux01:~$ cat secretmessage_ENCRYPTED.txt
U2FsdGVkX1+baXqJnQrUR3htHvUuDgBsdd6JKgmjpe80mPuQSV0cggVahwIQdcDL
qwVQZ0zBj2VfzJWNGpaesg==
student@TargetLinux01:~$
```

Using Encryption to Enhance Confidentiality and Integrity (4e)

Fundamentals of Information Systems Security, Fourth Edition - Lab 05

16. Make a screen capture showing the output of the ls command.



```
student@TargetLinux01:~$ touch secretmessage.txt
student@TargetLinux01:~$ echo "text" >> secretmessage.txt
student@TargetLinux01:~$ echo "Encrypted Message for Instructor" >> secretmessage.txt
student@TargetLinux01:~$ cat secretmessage.txt
text
Encrypted Message for Instructor
student@TargetLinux01:~$ openssl enc -aes-256-cbc -pbkdf2 -salt -a -e -in secretmessage.txt -out secretmessage_ENCRYPTED.txt
enter aes-256-cbc encryption password:
Verifying - enter aes-256-cbc encryption password:
student@TargetLinux01:~$ ls
Desktop  instructor_public.key  Public  Templates
Documents Music                secretmessage_ENCRYPTED.txt  Videos
Downloads Pictures            secretmessage.txt
student@TargetLinux01:~$ cat secretmessage_ENCRYPTED.txt
U2FsdGVkX1+baXq3nQrUR3htHvUuDgBsdd6JKgmjpe80mPuQSV0cggVaHwIQdcDL
qwVQZQzBj2VfzJWNGpaesg==
student@TargetLinux01:~$ rm secretmessage.txt
student@TargetLinux01:~$ ls
Desktop  instructor_public.key  Public  Videos
Documents Music                secretmessage_ENCRYPTED.txt
Downloads Pictures            Templates
student@TargetLinux01:~$
```

Part 3: Transfer and Decrypt a File Using Hybrid Cryptography

Using Encryption to Enhance Confidentiality and Integrity (4e)

Fundamentals of Information Systems Security, Fourth Edition - Lab 05

6. Make a screen capture showing the encrypted contents of the `secretkey_ENCRYPTED.txt` file.

```
Terminal - student@TargetLinux01
Using Encryption to Enhance Confidentiality and Integrity (4e)
TargetLinux01
2024-11-17 12:44:51
Brandon Trinkle
Jy0(010a000I00ob0,00{0^00]00
6
000`000FTw
:0Gh{0 d00(00.
B0003000@0 00|00EI|g000u:000x5^0v00{0sl;0G00k700K0000U^000 00b0>02[0000jk
0000000^;400{x0000u0000<k00m0A00[000000kI0Yd000Sy000J0
QP0M0r00X>*(000E04?x0j0!K0
v00340000x0fz^0000000qo9T00`8q00000P`H;00
7_000
00j0S080000$
."00j0=00Δ0K [0(0R00]X00000 P0?04/P4000.00s[0HHk#0000B@2fç0[0 Z000U\N.
,0-0FG,0V0z60UAR00e0K003gbiI0h=Z0;s0t0q}{30000kM0C00@
00FH0000:0
0"0b00(0v0'000800
D00k\l|[L0
^0000300600"b0<hy-0:0]0hb0zI00j0SfqSvd~60.0kp'(00>g0vb.00)sP~0[0]*gJ00V
Z000010I0#6|00$0H0+VGcJb#PQ000)00CH0^000000@G0J0`_0;
0^I00000H;xB0:0050\0(k0n00M000l0
.0t00z000K000m0y000000-p0Y0UYR000-L00:;{0:00007)!0,00h020w+!($80000{0x00000.7D0,
800 f0^0MB0<0n0>F0ZA0+000<x00>student@TargetLinux01:~$
```

Using Encryption to Enhance Confidentiality and Integrity (4e)

Fundamentals of Information Systems Security, Fourth Edition - Lab 05

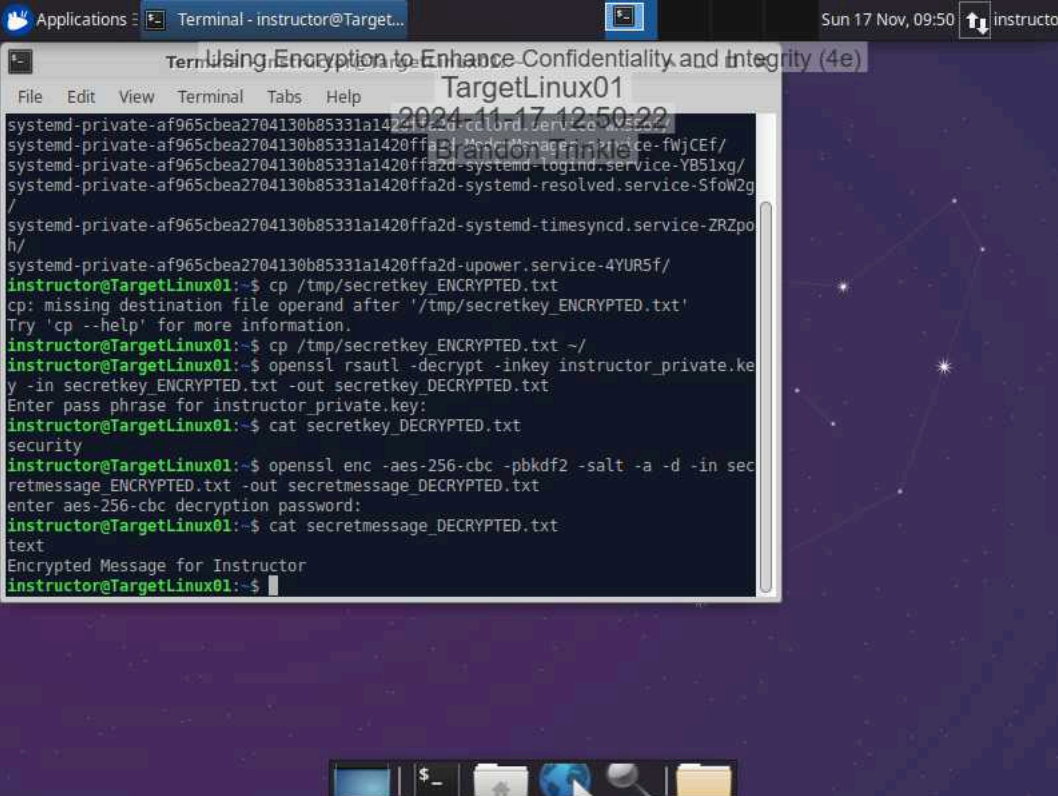
17. Make a screen capture showing the **decrypted contents of the secretkey_DECRYPTED.txt file.**

```
instructor@TargetLinux01:~$ cp /tmp/secretmessage_ENCRYPTED.txt /tmp/s
instructor@TargetLinux01:~$ cp /tmp/s
secretkey_ENCRYPTED.txt
secretmessage_ENCRYPTED.txt
ssh-gR4anHLPfokx/
systemd-private-af965cbea2704130b85331a1420ffa2d-colord.service-wX5Sof/
systemd-private-af965cbea2704130b85331a1420ffa2d-ModemManager.service-fwjCEf/
systemd-private-af965cbea2704130b85331a1420ffa2d-systemd-logind.service-YB51xg/
systemd-private-af965cbea2704130b85331a1420ffa2d-systemd-resolved.service-SfoW2g/
systemd-private-af965cbea2704130b85331a1420ffa2d-systemd-timesyncd.service-ZRZpo
h/
systemd-private-af965cbea2704130b85331a1420ffa2d-upower.service-4YUR5f/
instructor@TargetLinux01:~$ cp /tmp/secretkey_ENCRYPTED.txt
cp: missing destination file operand after '/tmp/secretkey_ENCRYPTED.txt'
Try 'cp --help' for more information.
instructor@TargetLinux01:~$ cp /tmp/secretkey_ENCRYPTED.txt ~/
instructor@TargetLinux01:~$ openssl rsautl -decrypt -inkey instructor_private.ke
y -in secretkey_ENCRYPTED.txt -out secretkey_DECRYPTED.txt
Enter pass phrase for instructor_private.key:
instructor@TargetLinux01:~$ cat secretkey_DECRYPTED.txt
security
instructor@TargetLinux01:~$
```

Using Encryption to Enhance Confidentiality and Integrity (4e)

Fundamentals of Information Systems Security, Fourth Edition - Lab 05

21. Make a screen capture showing the contents of the `secretmessage_DECRYPTED` file.



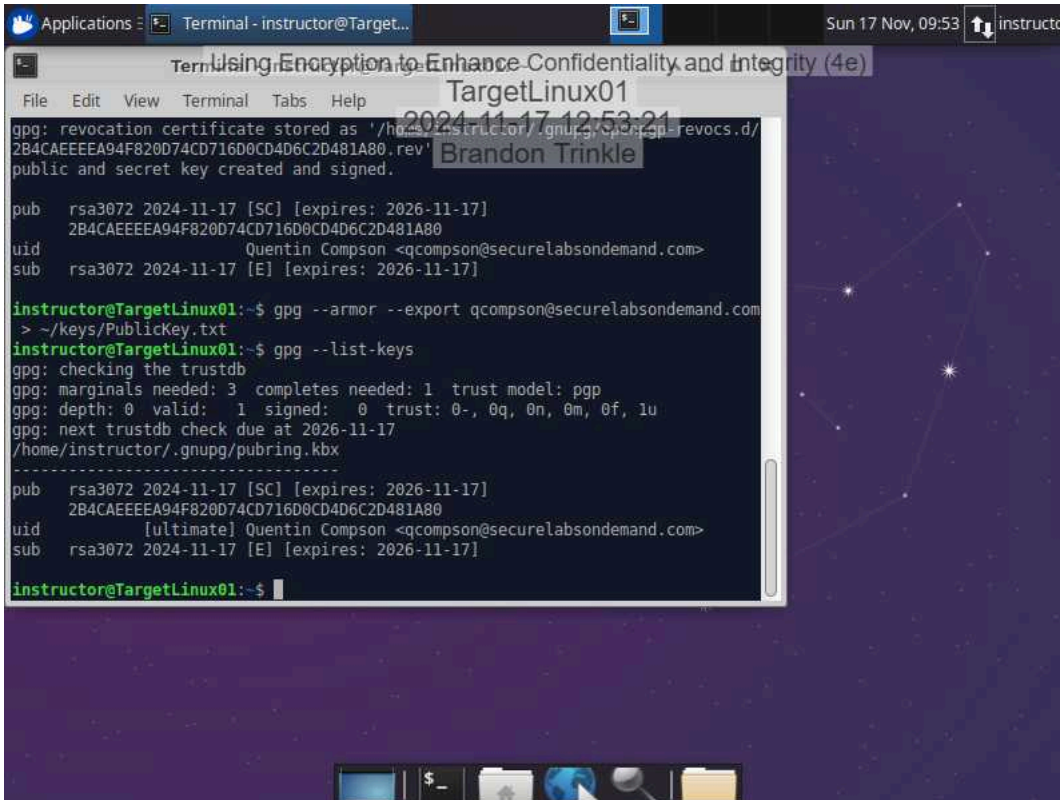
The screenshot shows a terminal window titled "Terminal - instructor@Target..." with a menu bar (File, Edit, View, Terminal, Tabs, Help). The terminal output shows the user navigating through system logs and then performing a decryption process. A timestamp "2024-11-17 12:50:22" and the name "Brandon Brinkley" are overlaid on the terminal. The decryption process involves copying a file, using a private key to decrypt it, and then using a password to decrypt the resulting file. The final output of the decryption is the text "Encrypted Message for Instructor".

```
systemd-private-af965cbea2704130b85331a1420ffa2d-systemd-resolved.service-SfoW2g/
systemd-private-af965cbea2704130b85331a1420ffa2d-systemd-logind.service-YB51xg/
systemd-private-af965cbea2704130b85331a1420ffa2d-systemd-resolved.service-SfoW2g/
/
systemd-private-af965cbea2704130b85331a1420ffa2d-systemd-timesyncd.service-ZRZpo
h/
systemd-private-af965cbea2704130b85331a1420ffa2d-upower.service-4YUR5f/
instructor@TargetLinux01:~$ cp /tmp/secretkey_ENCRYPTED.txt
cp: missing destination file operand after '/tmp/secretkey_ENCRYPTED.txt'
Try 'cp --help' for more information.
instructor@TargetLinux01:~$ cp /tmp/secretkey_ENCRYPTED.txt ~/
instructor@TargetLinux01:~$ openssl rsautl -decrypt -inkey instructor_private.ke
y -in secretkey_ENCRYPTED.txt -out secretkey_DECRYPTED.txt
Enter pass phrase for instructor_private.key:
instructor@TargetLinux01:~$ cat secretkey_DECRYPTED.txt
security
instructor@TargetLinux01:~$ openssl enc -aes-256-cbc -pbkdf2 -salt -a -d -in sec
retmessage_ENCRYPTED.txt -out secretmessage_DECRYPTED.txt
enter aes-256-cbc decryption password:
instructor@TargetLinux01:~$ cat secretmessage_DECRYPTED.txt
text
Encrypted Message for Instructor
instructor@TargetLinux01:~$
```

Section 3: Challenge and Analysis

Part 1: Digitally Sign a Document Using GPG

Make a screen capture showing the **key fingerprint** for the key pair you generated in this part of the lab.



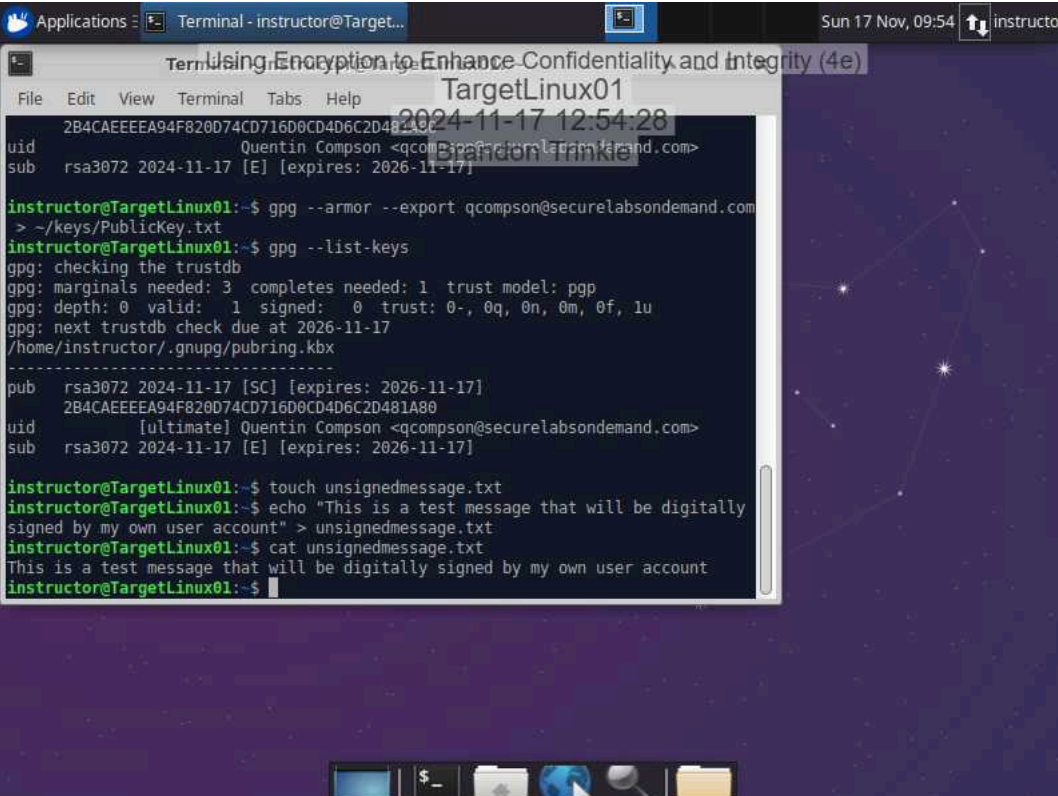
The screenshot shows a terminal window titled "Terminal - instructor@TargetLinux01" with a dark background and light text. The terminal output shows the successful generation of a GPG key pair and the subsequent listing of the keys. The key fingerprint is displayed as a long hexadecimal string. The terminal window is overlaid on a desktop background featuring a constellation of stars.

```
gpg: revocation certificate stored as '/home/instructor/.gnupg/openpgp-revocs.d/2B4CAEEEEA94F820D74CD716D0CD4D6C2D481A80.rev'  
public and secret key created and signed.  
  
pub  rsa3072 2024-11-17 [SC] [expires: 2026-11-17]  
     2B4CAEEEEA94F820D74CD716D0CD4D6C2D481A80  
uid           Quentin Compson <qcompson@securelabsondemand.com>  
sub  rsa3072 2024-11-17 [E] [expires: 2026-11-17]  
  
instructor@TargetLinux01:~$ gpg --armor --export qcompson@securelabsondemand.com > ~/keys/PublicKey.txt  
instructor@TargetLinux01:~$ gpg --list-keys  
gpg: checking the trustdb  
gpg: marginals needed: 3  completes needed: 1  trust model: pgp  
gpg: depth: 0  valid: 1  signed: 0  trust: 0-, 0q, 0n, 0m, 0f, 1u  
gpg: next trustdb check due at 2026-11-17  
/home/instructor/.gnupg/pubring.kbx  
-----  
pub  rsa3072 2024-11-17 [SC] [expires: 2026-11-17]  
     2B4CAEEEEA94F820D74CD716D0CD4D6C2D481A80  
uid           [ultimate] Quentin Compson <qcompson@securelabsondemand.com>  
sub  rsa3072 2024-11-17 [E] [expires: 2026-11-17]  
  
instructor@TargetLinux01:~$
```

Using Encryption to Enhance Confidentiality and Integrity (4e)

Fundamentals of Information Systems Security, Fourth Edition - Lab 05

Make a screen capture showing the contents of the unsignedmessage.txt file.

A screenshot of a Linux terminal window titled 'Terminal - instructor@TargetLinux01'. The window shows the output of several GPG commands. At the top, a GPG key is displayed with its fingerprint and user information. Below this, the user runs 'gpg --armor --export qcompson@securelabsondemand.com' to export the key to a file. Then, they run 'gpg --list-keys' which shows the key details. Finally, they create a file 'unsignedmessage.txt' with the text 'This is a test message that will be digitally signed by my own user account' and then display its contents with 'cat unsignedmessage.txt'. The terminal output is as follows:

```
2B4CAEEEEAA94F820D74CD716D0CD4D6C2D481A80
uid      Quentin Compson <qcompson@securelabsondemand.com>
sub      rsa3072 2024-11-17 [E] [expires: 2026-11-17]

instructor@TargetLinux01:~$ gpg --armor --export qcompson@securelabsondemand.com
> ~/keys/PublicKey.txt
instructor@TargetLinux01:~$ gpg --list-keys
gpg: checking the trustdb
gpg: marginals needed: 3 completes needed: 1 trust model: pgp
gpg: depth: 0 valid: 1 signed: 0 trust: 0-, 0q, 0n, 0m, 0f, 1u
gpg: next trustdb check due at 2026-11-17
/home/instructor/.gnupg/pubring.kbx
-----
pub      rsa3072 2024-11-17 [SC] [expires: 2026-11-17]
         2B4CAEEEEAA94F820D74CD716D0CD4D6C2D481A80
uid      [ultimate] Quentin Compson <qcompson@securelabsondemand.com>
sub      rsa3072 2024-11-17 [E] [expires: 2026-11-17]

instructor@TargetLinux01:~$ touch unsignedmessage.txt
instructor@TargetLinux01:~$ echo "This is a test message that will be digitally
signed by my own user account" > unsignedmessage.txt
instructor@TargetLinux01:~$ cat unsignedmessage.txt
This is a test message that will be digitally signed by my own user account
instructor@TargetLinux01:~$
```

Part 2: Verify the Digital Signature Using Kleopatra

Using Encryption to Enhance Confidentiality and Integrity (4e)

Fundamentals of Information Systems Security, Fourth Edition - Lab 05

Make a screen capture showing the successful signature verification on the signed message file.

