

Using Encryption to Enhance Confidentiality and Integrity (4e)

Fundamentals of Information Systems Security, Fourth Edition - Lab 05

Student:

Kanchan Chopde

Email:

hq0656@wayne.edu

Time on Task:

2 hours, 26 minutes

Progress:

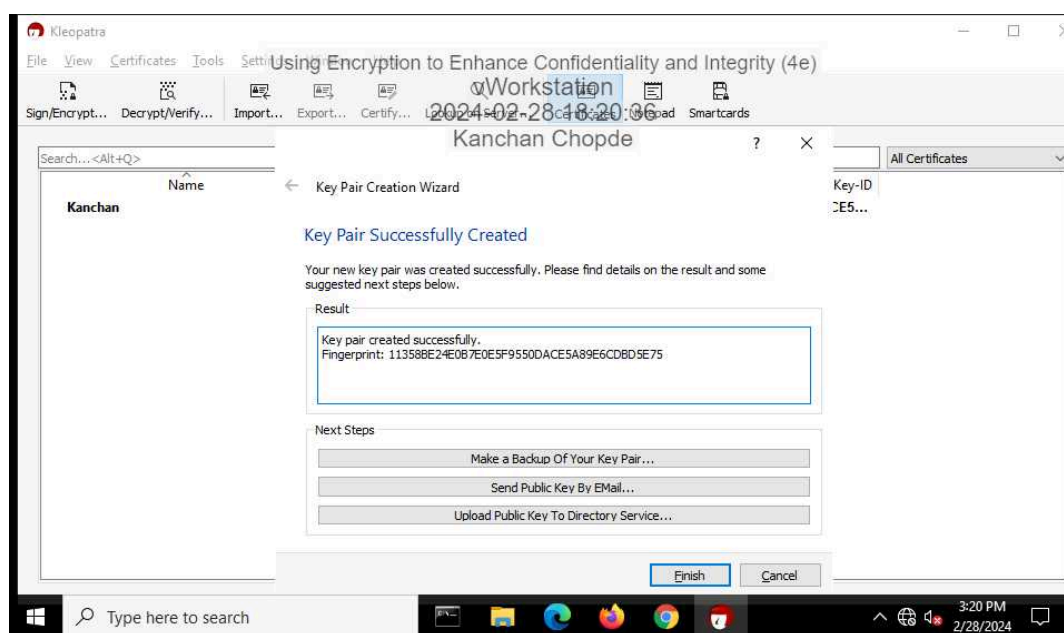
100%

Report Generated: Thursday, February 29, 2024 at 9:33 AM

Section 1: Hands-On Demonstration

Part 1: Create and Exchange Asymmetric Encryption Keys

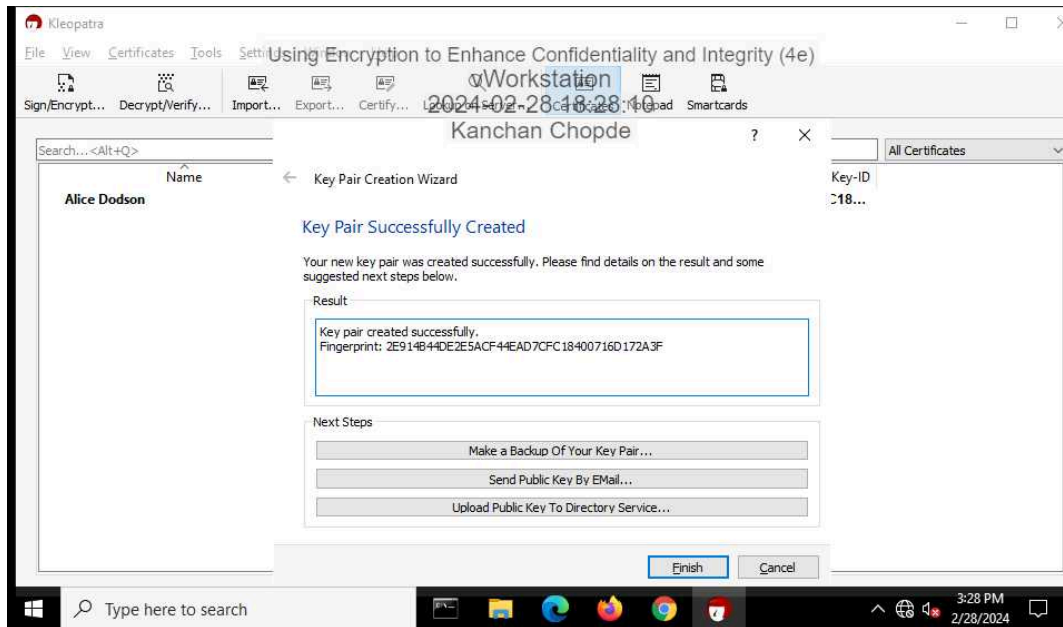
9. Make a screen capture showing the **fingerprint** for your key pair.



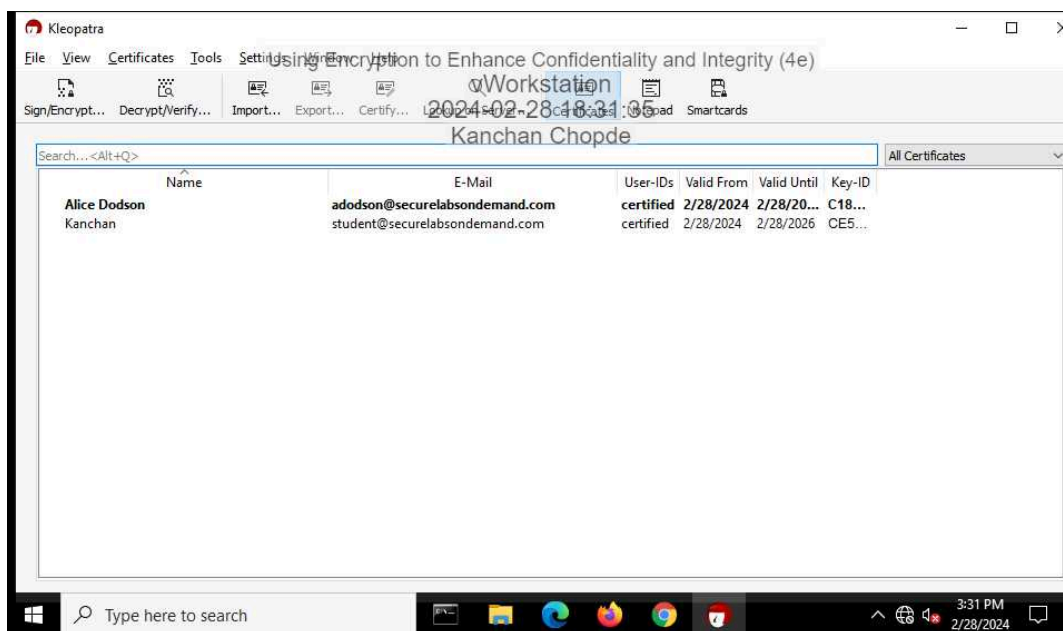
Using Encryption to Enhance Confidentiality and Integrity (4e)

Fundamentals of Information Systems Security, Fourth Edition - Lab 05

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



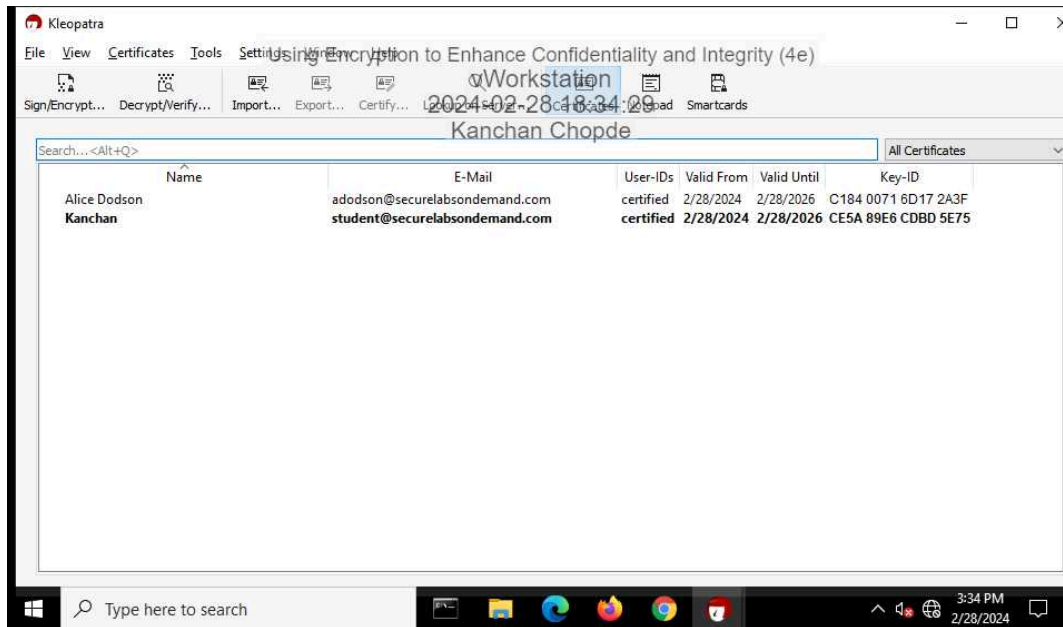
30. Make a screen capture showing your public key in Alice's certificate cache.



Using Encryption to Enhance Confidentiality and Integrity (4e)

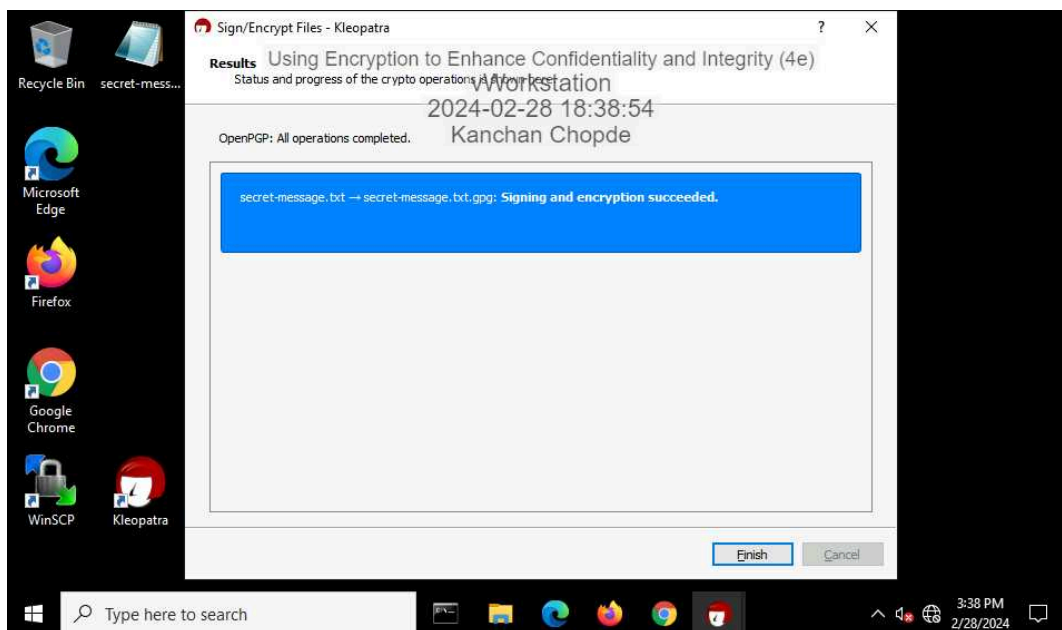
Fundamentals of Information Systems Security, Fourth Edition - Lab 05

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



Part 2: Encrypt a File Using Asymmetric Encryption

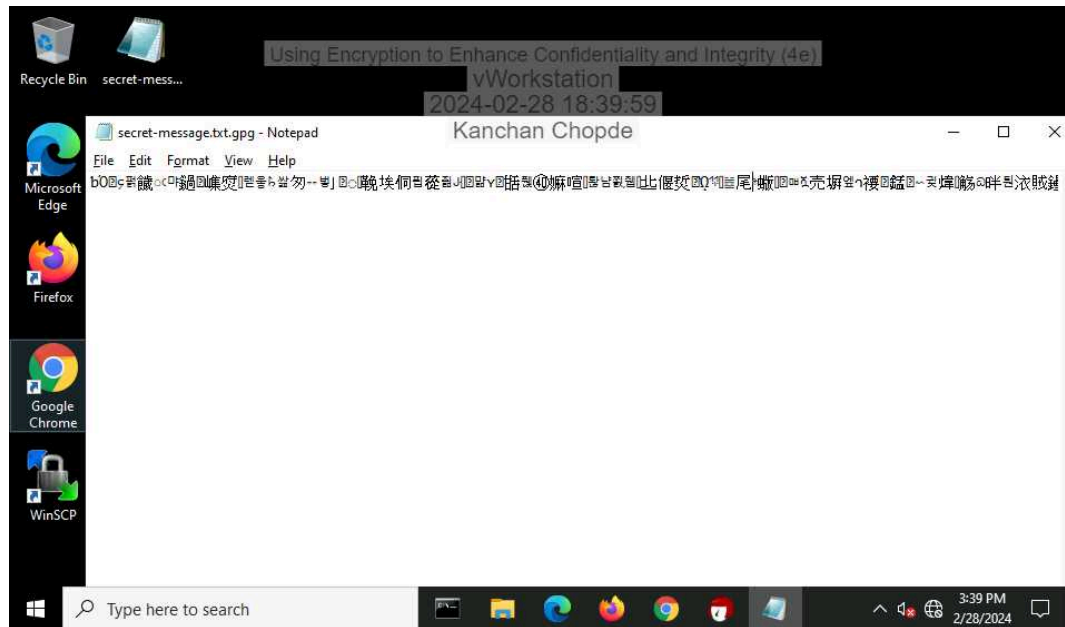
9. Make a screen capture showing the successful signing and encryption message.



Using Encryption to Enhance Confidentiality and Integrity (4e)

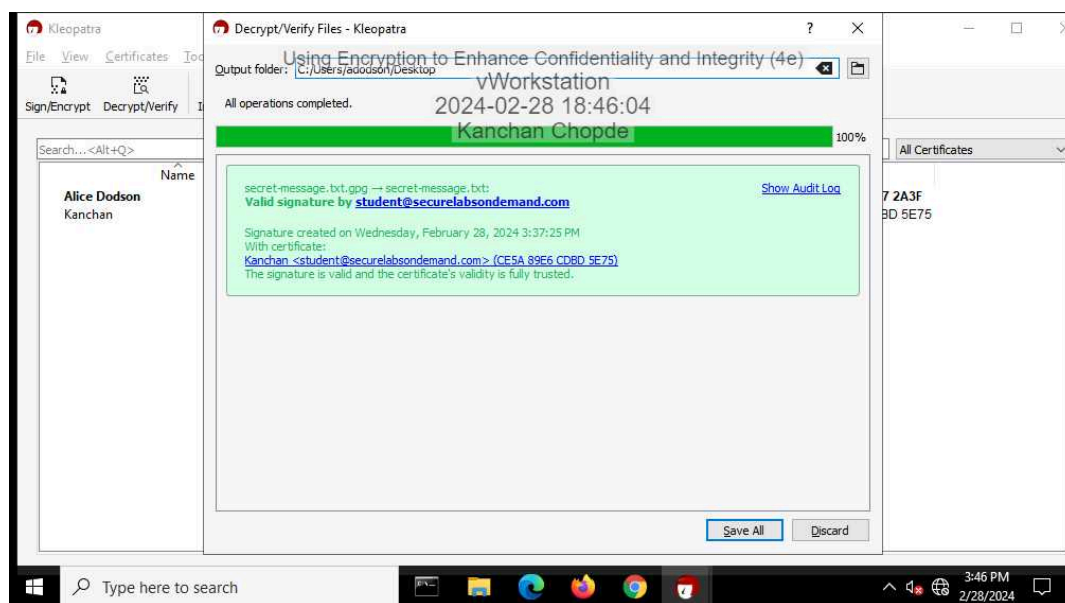
Fundamentals of Information Systems Security, Fourth Edition - Lab 05

12. Make a screen capture showing the **ciphertext**.



Part 3: Decrypt a File Using Asymmetric Encryption

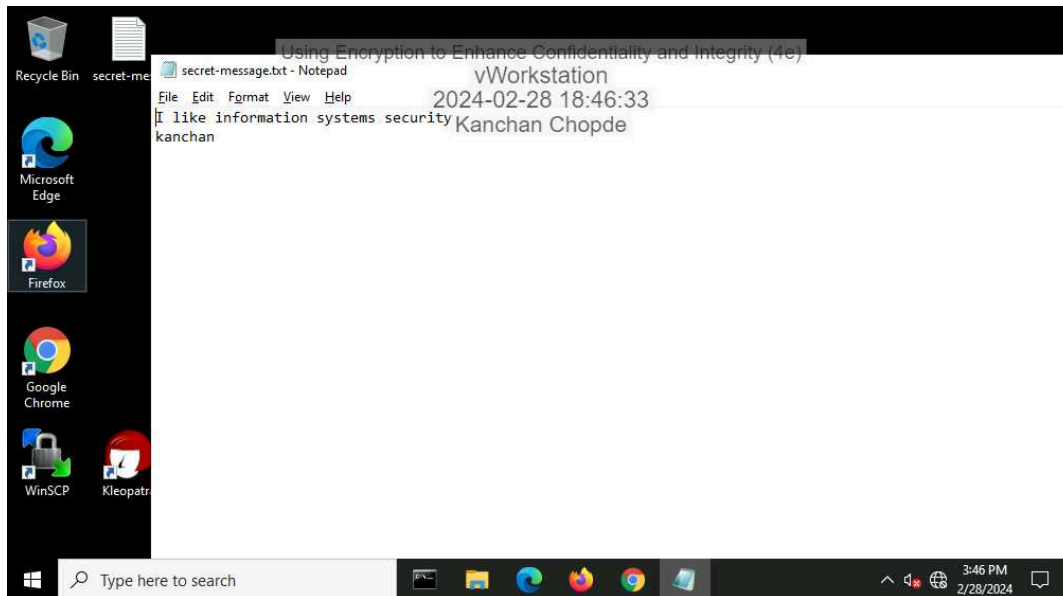
15. Make a screen capture showing the **Decrypt/Verify Files** window.



Using Encryption to Enhance Confidentiality and Integrity (4e)

Fundamentals of Information Systems Security, Fourth Edition - Lab 05

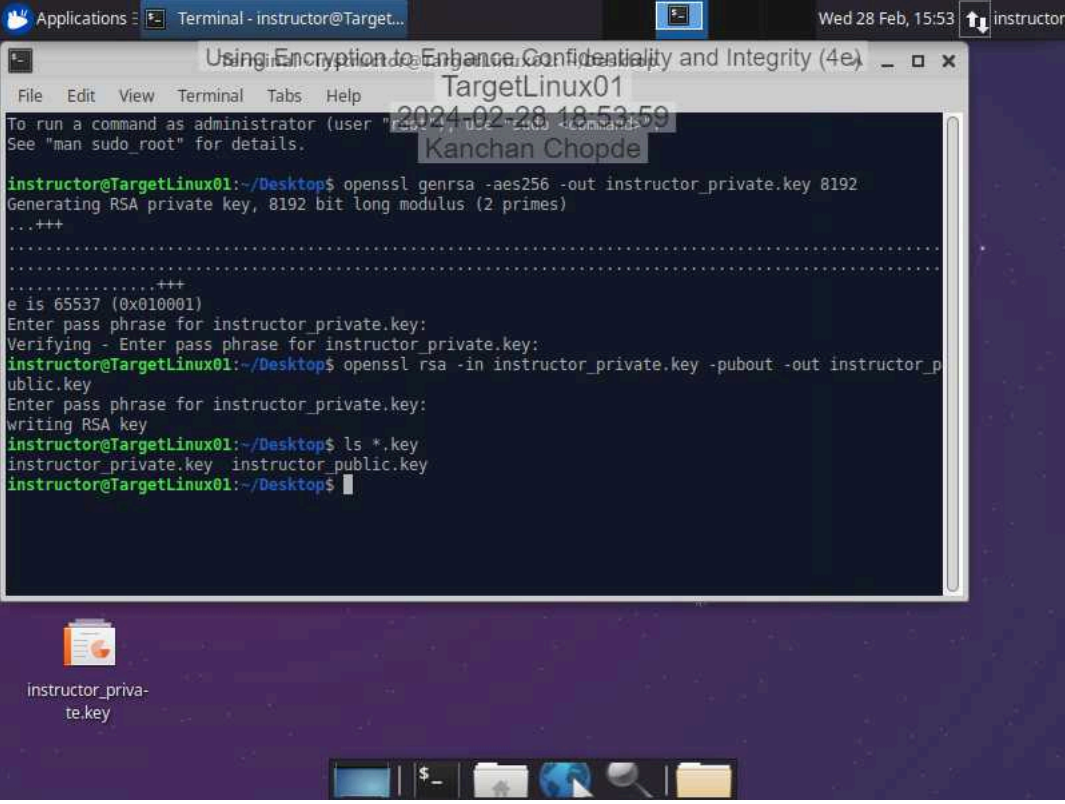
18. Make a screen capture showing the **decrypted secret-message.txt** file in Notepad.



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..." on a Linux desktop. The terminal displays the following commands and output:

```
instructor@TargetLinux01:~/Desktop$ openssl genrsa -aes256 -out instructor_private.key 8192
Generating RSA private key, 8192 bit long modulus (2 primes)
.....+++
.....+++
e is 65537 (0x010001)
Enter pass phrase for instructor_private.key:
Verifying - Enter pass phrase for instructor_private.key:
instructor@TargetLinux01:~/Desktop$ openssl rsa -in instructor_private.key -pubout -out instructor_public.key
Enter pass phrase for instructor_private.key:
writing RSA key
instructor@TargetLinux01:~/Desktop$ ls *.key
instructor_private.key  instructor_public.key
instructor@TargetLinux01:~/Desktop$
```

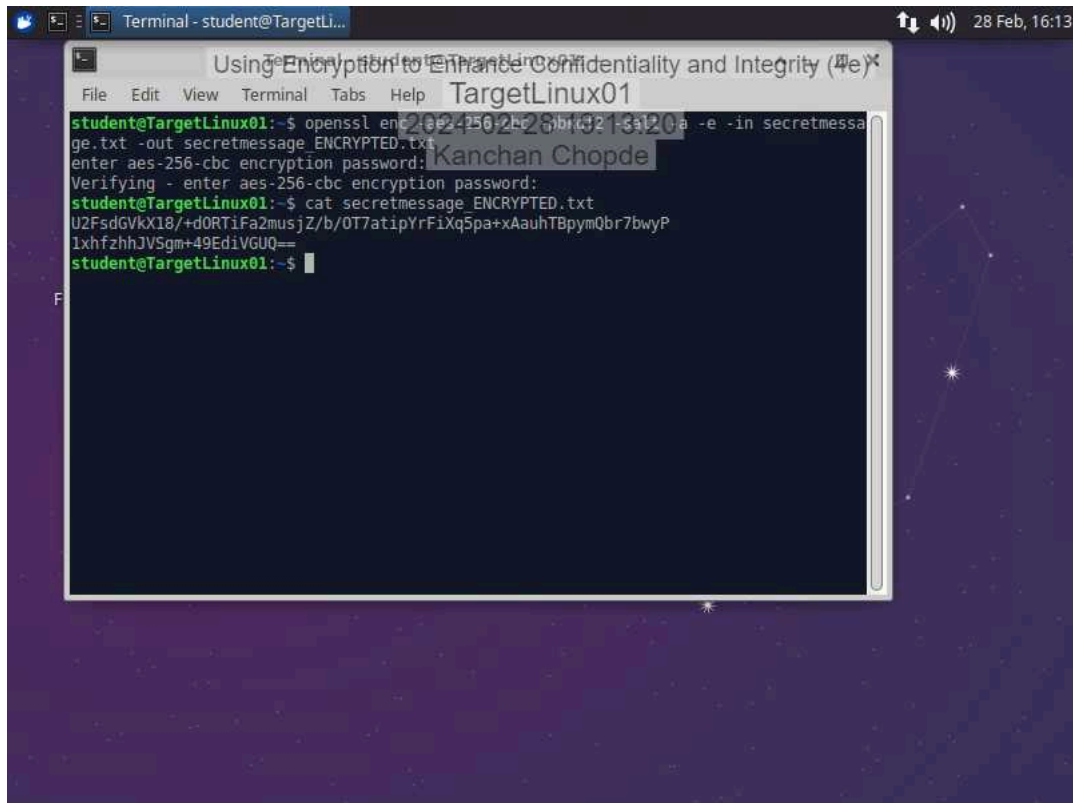
Below the terminal window, a file icon for "instructor_private.key" is visible on the desktop. The desktop background is a dark purple pattern. The terminal window has a menu bar with "File", "Edit", "View", "Terminal", "Tabs", and "Help". The desktop taskbar at the bottom shows icons for a terminal, a file manager, a web browser, and a search icon.

Part 2: Encrypt a File Using Symmetric Encryption

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

yourpassword

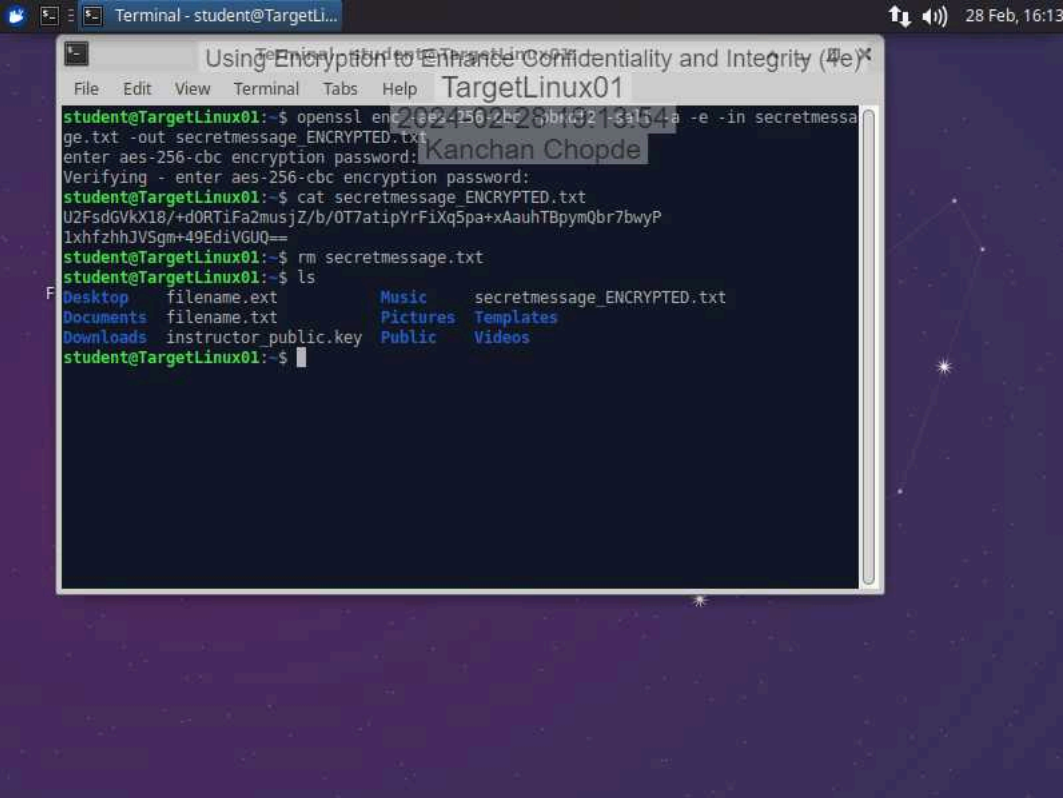
13. Make a screen capture showing the ciphertext in the `secretmessage_ENCRYPTED.txt` file.



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.



A terminal window titled "Terminal - student@TargetLinux01" is shown. The window has a menu bar with "File", "Edit", "View", "Terminal", "Tabs", and "Help". The terminal output shows the following commands and their results:

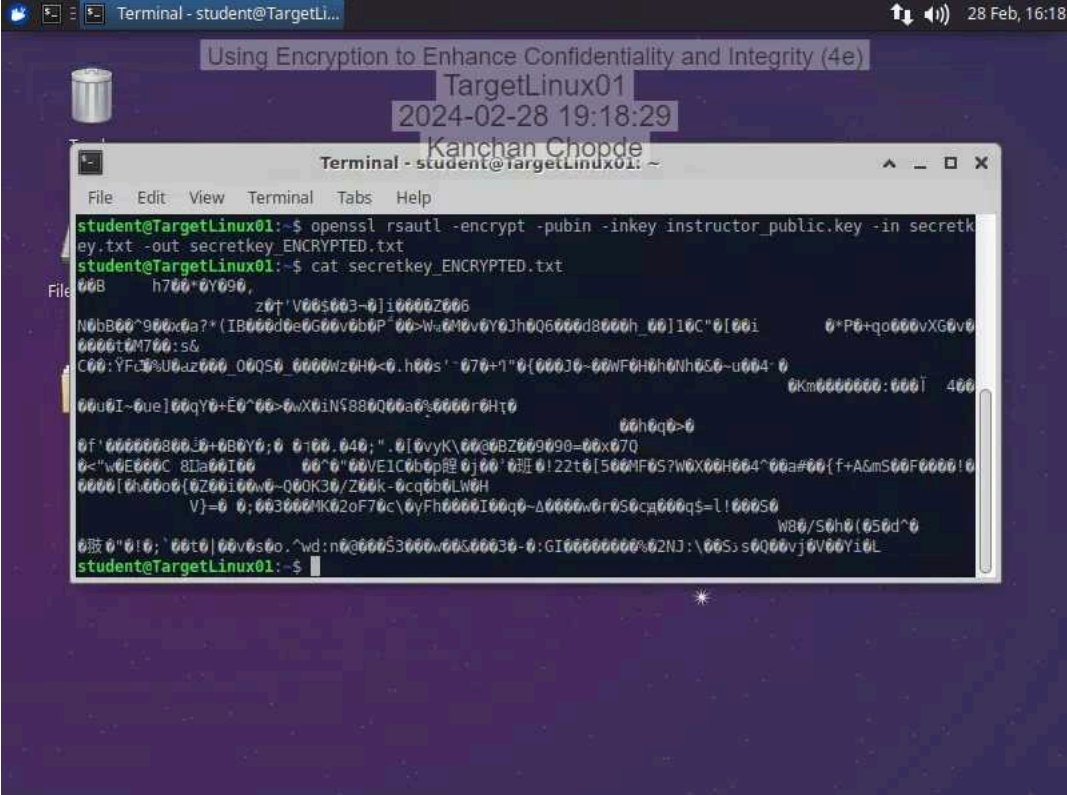
```
student@TargetLinux01:~$ openssl enc aes-256-cbc -pbkdf2 -salt -a -e -in secretmessage.txt -out secretmessage_ENCRYPTED.txt
enter aes-256-cbc encryption password: Kanchan Chopde
Verifying - enter aes-256-cbc encryption password:
student@TargetLinux01:~$ cat secretmessage_ENCRYPTED.txt
U2FsdGvKX18/+dORTiFa2musjZ/b/OT7atipYrFiXq5pa+xAauhTBpymQbr7bwyP
1xhfzhhJVSgm+49EdiVGUQ==
student@TargetLinux01:~$ rm secretmessage.txt
student@TargetLinux01:~$ ls
Desktop      filename.ext      Music      secretmessage_ENCRYPTED.txt
Documents    filename.txt      Pictures   Templates
Downloads    instructor_public.key  Public     Videos
```

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.



The screenshot shows a terminal window titled "Terminal - student@TargetLinux01" with a dark purple background. The terminal displays the following commands and output:

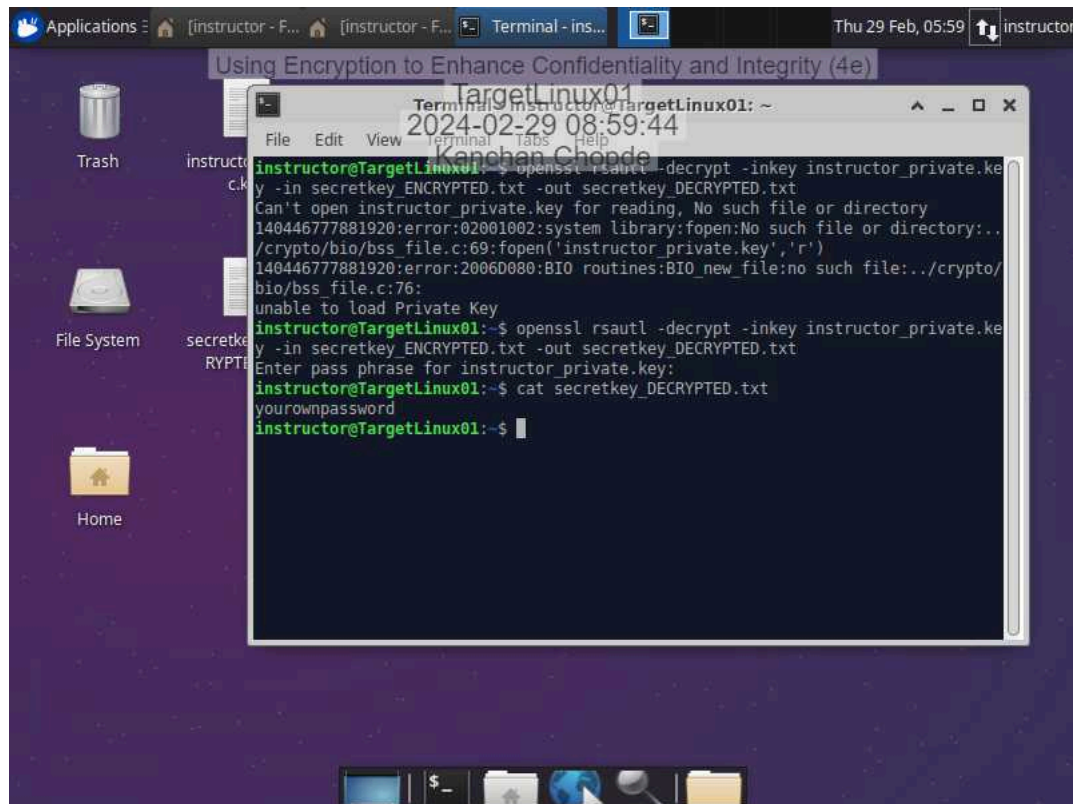
```
student@TargetLinux01:~$ openssl rsautl -encrypt -pubin -inkey instructor_public.key -in secretkey.txt -out secretkey_ENCRYPTED.txt
student@TargetLinux01:~$ cat secretkey_ENCRYPTED.txt
```

The output of the `cat` command is a large block of base64-encoded text, which is the encrypted contents of the `secretkey.txt` file. The text is displayed in a monospaced font and is partially obscured by a vertical scrollbar on the right side of the terminal window.

Using Encryption to Enhance Confidentiality and Integrity (4e)

Fundamentals of Information Systems Security, Fourth Edition - Lab 05

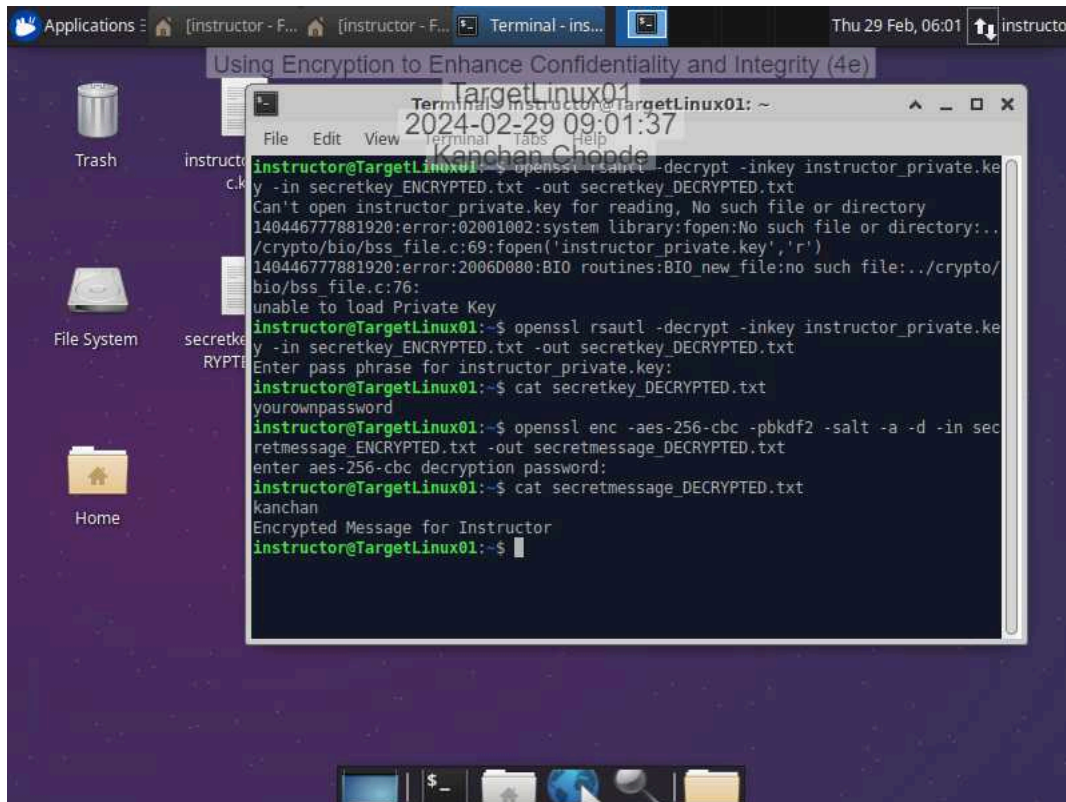
17. Make a screen capture showing the **decrypted contents of the secretkey_DECRYPTED.txt** file.



Using Encryption to Enhance Confidentiality and Integrity (4e)

Fundamentals of Information Systems Security, Fourth Edition - Lab 05

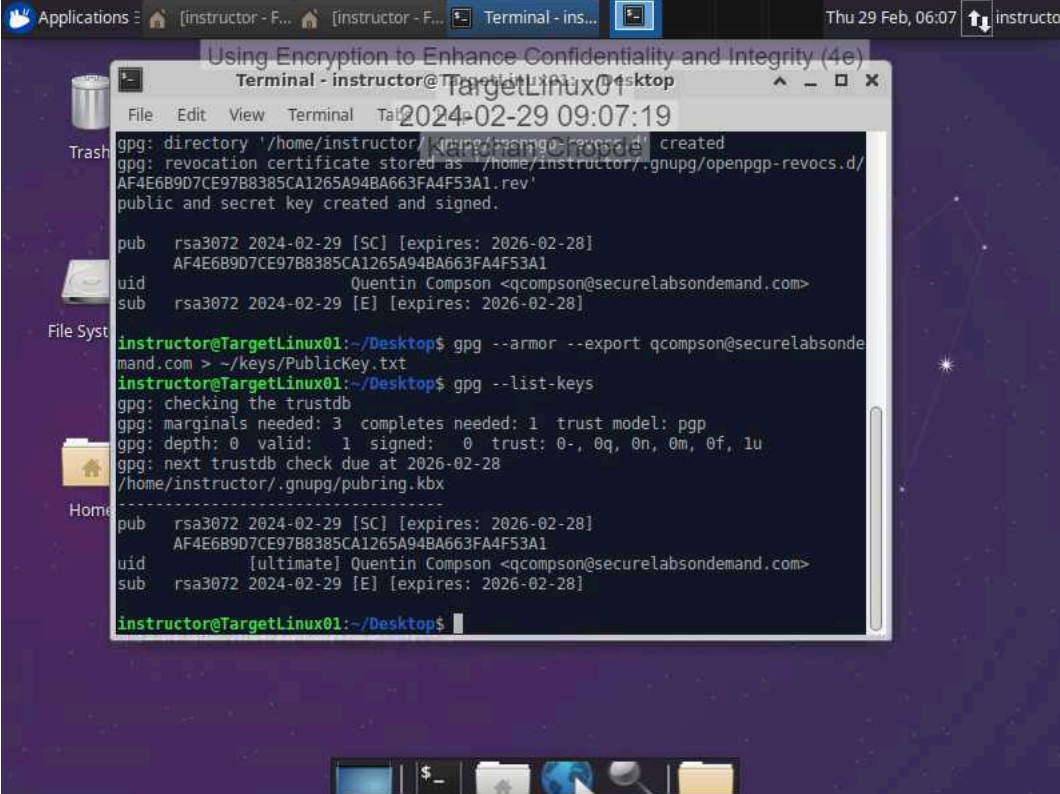
21. Make a screen capture showing the contents of the `secretmessage_DECRYPTED` file.



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 Linux desktop environment with a terminal window open. The terminal displays the output of several GPG commands. The first command, `gpg --full-genkey`, generates a new key pair and stores the revocation certificate. The second command, `gpg --armor --export qcompson@securelabsondemand.com > ~/keys/PublicKey.txt`, exports the public key to a file. The third command, `gpg --list-keys`, lists the keys in the trust database, showing the key fingerprint and expiration date. The terminal output is as follows:

```
gpg: directory '/home/instructor/.gnupg/openpgp-revocs.d/' created
gpg: revocation certificate stored as '/home/instructor/.gnupg/openpgp-revocs.d/AF4E6B9D7CE97B8385CA1265A94BA663FA4F53A1.rev'
public and secret key created and signed.

pub  rsa3072 2024-02-29 [SC] [expires: 2026-02-28]
     AF4E6B9D7CE97B8385CA1265A94BA663FA4F53A1
uid                               Quentin Compson <qcompson@securelabsondemand.com>
sub  rsa3072 2024-02-29 [E] [expires: 2026-02-28]

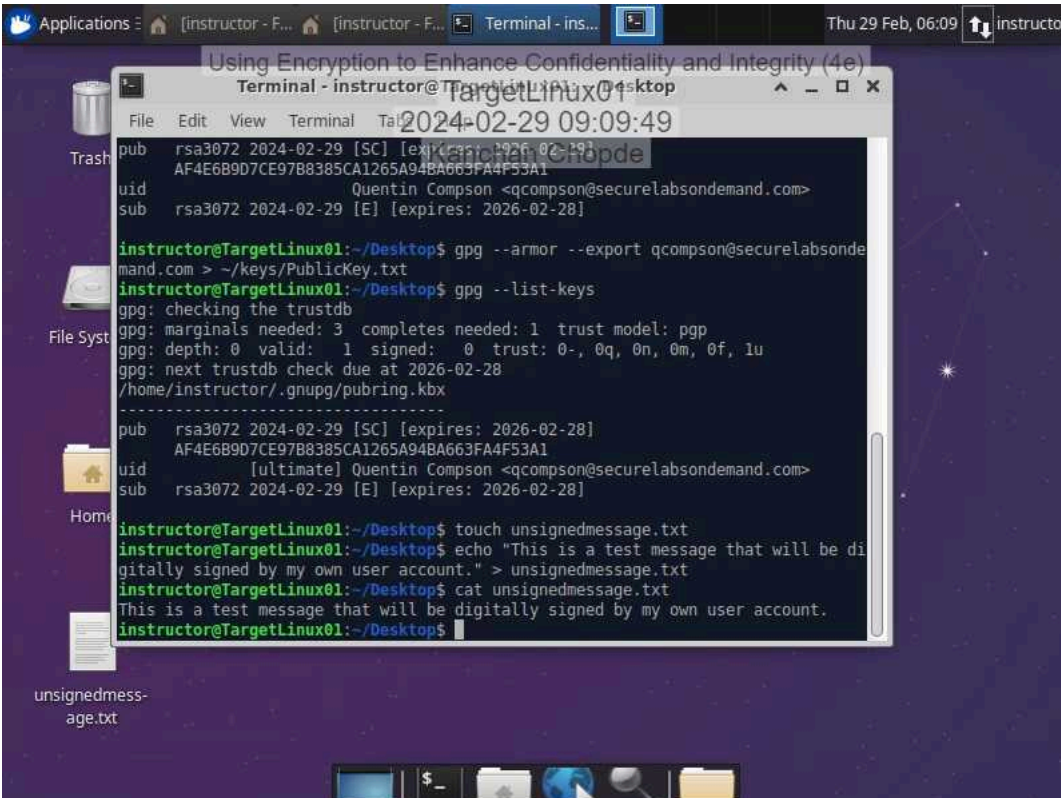
instructor@TargetLinux01:~/Desktop$ gpg --armor --export qcompson@securelabsondemand.com > ~/keys/PublicKey.txt
instructor@TargetLinux01:~/Desktop$ gpg --list-keys
gpg: checking the trustdb
gpg: marginals needed: 3 completes needed: 1 trust model: pgp
gpg: depth: 0 valid: 1 signed: 0 trust: 0-, 0q, 0n, 0m, 0f, 1u
gpg: next trustdb check due at 2026-02-28
/home/instructor/.gnupg/pubring.kbx
-----
pub  rsa3072 2024-02-29 [SC] [expires: 2026-02-28]
     AF4E6B9D7CE97B8385CA1265A94BA663FA4F53A1
uid                               [ultimate] Quentin Compson <qcompson@securelabsondemand.com>
sub  rsa3072 2024-02-29 [E] [expires: 2026-02-28]

instructor@TargetLinux01:~/Desktop$
```

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.



The screenshot shows a Linux desktop with a terminal window open. The terminal displays the following commands and output:

```
instructor@TargetLinux01:~/Desktop$ gpg --armor --export qcompson@securelabsondemand.com > ~/keys/PublicKey.txt
instructor@TargetLinux01:~/Desktop$ 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 2026-02-28
/home/instructor/.gnupg/pubring.kbx
-----
pub   rsa3072 2024-02-29 [SC] [expires: 2026-02-28]
       AF4E6B9D7CE97B8385CA1265A948A663FA4F53A1
uid   [ultimate] Quentin Compson <qcompson@securelabsondemand.com>
sub   rsa3072 2024-02-29 [E] [expires: 2026-02-28]

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

The desktop background is a dark purple space-themed wallpaper. The terminal window has a title bar that reads "Terminal - instructor@TargetLinux01:~/Desktop". The system clock in the top right corner shows "Thu 29 Feb, 06:09".

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.

