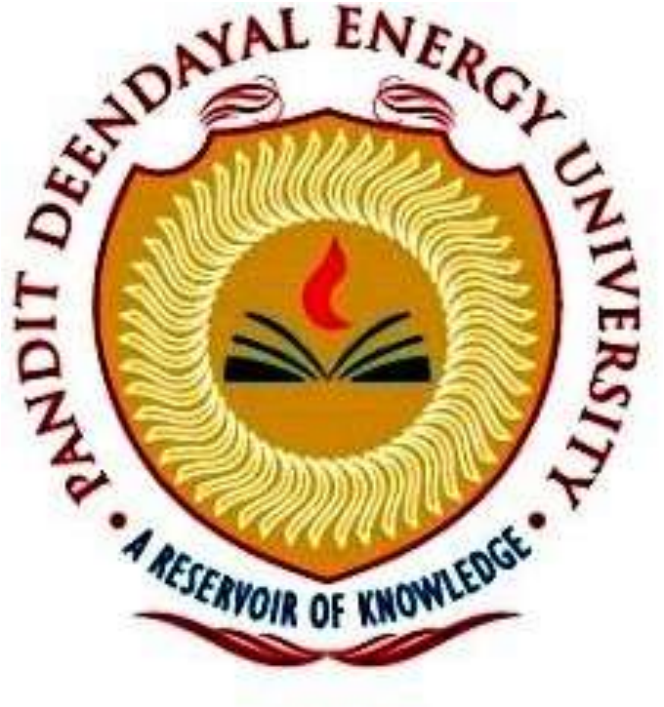
Information Security Lab

(20CP304P)

Smit Sutariya

Roll no. 21BCP142

Div:3 G:5



Faculty Name: Hargeet Kaur

COMPUTER ENGINEERING

School of Technology,

Pandit Deendayal

EnergyUniversity

January – 2023

EXPERIMENT: 1

Aim: Download and practice cryptool.

Introduction:

➔CrypTool is an open-source, cross-platform software application for cryptography education and research. It is used to illustrate cryptographic and cryptanalytic concepts, and to experiment with different algorithms and techniques.

➔CrypTool includes a wide variety of features, including:

- The Workspace Manager: The Workspace Manager is a graphical programming environment that allows you to create your own cryptographic applications. You can use the Workspace Manager to experiment with different algorithms and techniques, and to create custom tools for your own needs.

- The Templates: CrypTool includes a library of templates that you can use to encrypt and decrypt text, generate digital signatures, and crack passwords. The templates are a great way to learn about different cryptographic concepts without having to write any code yourself.

- The Help System: The CrypTool help system provides detailed information on all of the features of CrypTool. It is a great resource for learning about cryptography, and for troubleshooting problems.

➔CrypTool is a valuable tool for anyone who wants to learn about cryptography. It is used by students, researchers, and security professionals around the world.

Application:

1. Education:

➔In an educational setting, CrypTool can be used to teach students about the following topics:

- The basics of cryptography, such as encryption, decryption, and digital signatures.
- Different types of cryptographic algorithms, such as block ciphers, stream ciphers, and public-key cryptography.
- The security of cryptographic systems.

2. Research:

➔In a research setting, CrypTool can be used to do the following:

- Test new cryptographic algorithms and techniques.
- Analyze the security of existing cryptographic systems.
- Develop new tools for breaking cryptographic systems.

3. Security:

➔In a security setting, CrypTool can be used to do the following:

- Encrypt files, emails, and passwords.
- Protect sensitive information from unauthorized access.
- Detect and prevent cyberattacks.

4. Entertainment:

➔In an entertainment setting, CrypTool can be used to do the following:

- Create puzzles and challenges.
- Play games.
- Learn about cryptography in a fun and engaging way.

Reference:

➔ https://www.guru99.com/how-to-make-your-data-safe-using-cryptography.html

➔ https://bard.google.com/

Experiment-2

Aim: Study and Implement program for Caeser Cipher with Encryption, Decryption, Brute Force Attack, and Frequency Analysis functions.

Introduction:

In the realm of information security and cryptography, the Caesar Cipher stands as a venerable testament to the origins of encryption. Named after the ancient Roman military leader Julius Caesar, this cipher exemplifies a basic substitution technique within the broader scope of cryptographic methods.

At its core, the Caesar Cipher operates on a fundamental principle: each letter in the plaintext is shifted a fixed number of positions down or up the alphabet to generate the corresponding ciphertext. This technique, classified as a symmetric-key substitution cipher, underscores the concept of confidentiality by obfuscating the original message's content.

While modern cryptography has evolved exponentially, comprehending the mechanics of historic techniques like the Caesar Cipher remains pivotal. This lab manual ventures into the practicalities of this cipher, providing step-by-step directives for both encryption and decryption procedures. By manipulating the parameters of the Caesar Cipher, you'll gain insights into its strengths, weaknesses, and vulnerabilities

Source Code:

```
def encrypt(text, key):
    cipher = ''
    for char in text:
        if char == ' ':
            cipher += ''
        elif char.isupper():
            cipher += chr((ord(char) + key - 65) % 26 + 65)
        else:
            cipher += chr((ord(char) + key - 97) % 26 + 97)
    return cipher


def decrypt(text, key):
    cipher = ''
    for char in text:
        if char == ' ':
            cipher += ''
        elif char.isupper():
            cipher += chr((ord(char) - key - 65) % 26 + 65)
        else:
            cipher += chr((ord(char) - key - 97) % 26 + 97)
    return cipher

def main():
    p_text = input("Enter text to encrypt: ")
    key = int(input("Enter key: "))
    print("Plain text: ", p_text)
    e_text = encrypt(p_text,key)
    print("encrypted text: ", e_text)
    print("decrypted text: ", decrypt(e_text,key))

if __name__=="__main__":
    main()
```

Outputs:

Program output:

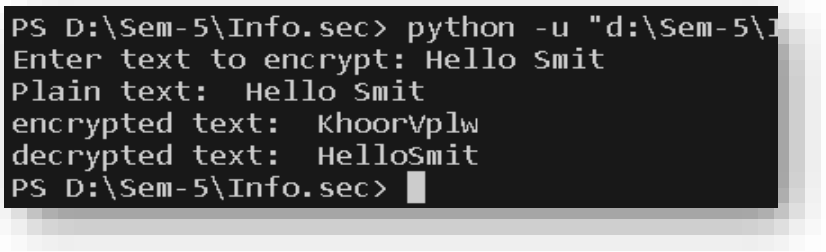
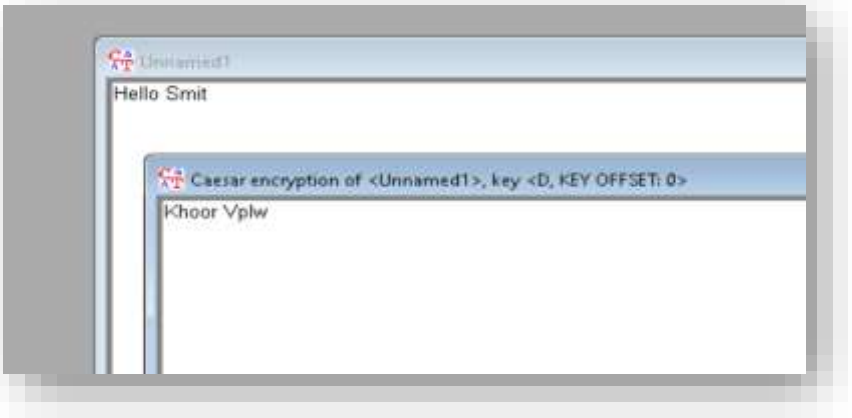


Fig. 1.1

CrypTool Output:



1.2 Encryption

1.3 Decryption

Cryptanalysis:

1. Brute Force Attack: In a brute force attack, you try all possible keys (shift values) until you find the one that successfully decrypts the ciphertext into meaningful plaintext. Since the Caesar cipher only has 25 possible keys (since a shift of 0 results in the original plaintext), this method is easily applicable.
2. Frequency Analysis: Frequency analysis involves analyzing the frequency of letters in the ciphertext. In English, certain letters and combinations of letters appear more frequently than others. For example, 'E' is the most common letter. By analyzing the frequency distribution of letters in the ciphertext, you can try to deduce the most likely shift used in the cipher.
3. Automated Tools: There are various online tools and software available that can perform cryptanalysis of the Caesar cipher automatically. These tools use algorithms and techniques to quickly determine the most likely shift value.

Application:

- As a simple method to add basic encryption to messages. For example, Julius Caesar is said to have used it to communicate with his generals.
- As a teaching tool to illustrate encryption concepts. It provides a simple substitution cipher to demonstrate core encryption principles.
- To obscure words or passages in media to avoid filters or spoiling plot points.
- To provide a layer of simple obscurity for things like website or product codes.

References:

javaTpoint: https://www.javatpoint.com/caesar-cipher-technique
openai: https://www.chat.openai.com