LAB NO: 4/5 **Date:**

CRYPTOGRAPHIC ALGORITHMS

Objectives:

In this lab, student will be able to:

- Identify the working of cryptographic algorithms
- Identify the need for the same.
- Identify the different types of cryptosystem
- Implement the algorithms

Description:

Cryptography is the study of encrypting and decrypting data to prevent unauthorized access. The ciphertext should be known by both the sender and the recipient. With the advancement of modern data security, we can now change our data such that only the intended recipient can understand it. Cryptography allows for the **secure transmission** of digital data between willing parties. It is used to safeguard company secrets, secure classified information, and sensitive information from fraudulent activity, among other things. Crypto means hidden and graph means writing.

Note:

1. **The language of implementation is up to the student (C,Java,Python)**
2. **The students should not use the built in functions or libraries or API for the encryption/decryption steps**

LAB EXERCISES

1. Implement the following algorithms and identify the time taken for encryption and decryption using different size of plain text (plain text should be in a file) and plot a graph for each of the algorithm given (Time Vs File size,Time VS key length)

    a) S-DES Algorithm
    b) DES Algorithm
    c) AES Algorithm
    d) Diffie hellman Algorithm
    e) RSA
    f) ECC

2. Generate Self-Signed SSL Certificate with OPENSSL in Kali Linux

    (refer Link:
    Generate Self-Signed SSL Certificate with OPENSSL in Kali Linux - Yeah Hub
    How to generate a self-signed SSL certificate on Linux - Linux Tutorials - Learn Linux Configuration.
)

LAB NO: 6 **Date:**

HASHCAT: (tool for penetration testers/ Password strength)

Objectives:

In this lab, student will be able to:

- Use the tool named Hashcat
- Identify the working of the tool
- Perform penetration testing
- Identify the strength of the password

Description:
Hashcat is a popular and effective password cracker widely used by both penetration testers and sysadmins as well as criminals and spies.Cracking passwords is different from guessing a web login password, which typically only allows a small number of guesses before locking your account. Instead, someone who has gained access to a system with encrypted passwords ("hashes") will often try to crack those hashes to recover those passwords.Passwords are no longer stored in plaintext (or shouldn't be, anyway). Instead, passwords are encrypted using a one-way function called a hash. Calculating a password like "Password1" into a hash is lightning quick. What if all you've got is the hash? A brute-force attack to reverse the hash function and recover the password could be computationally infeasible. Like, until the heat death of the universe infeasible.Luckily, or unluckily depending on your point of view, none of us is likely to live that long, but there are many ways to reverse a hash to recover the original password without resorting to a probably fruitless brute-force attack.

Solved Exercise

1. Hashcat's help menu using this command:

```
hashcat -h
```

LAB EXERCISES

1. Demonstrate the working of **Hashcat tool.**
2. Demonstrate the following attacks using the Hashcat tool

   Brute-Force attack Combinator attack Dictionary attack
   Fingerprint attack Hybrid attack Mask attack
   Permutation attack Rule-based attack Table-Lookup attack
   Toggle-Case attack PRINCE attack

Note : the link for reference is as follows:

https://www.freecodecamp.org/news/hacking-with-hashcat-a-practical-guide/
hashcat | Kali Linux Tools.
Hashcat explained: How this password cracker works | CSO Online.