**MAC-Message Authentication Codes(using colab)**

**Code:-**

import hashlib

import hmac

def generate\_mac(key, message):

    # Use SHA-256 as the hash function, you can choose a different one if needed

    hash\_function = hashlib.sha256

    # Generate the MAC using HMAC

    mac = hmac.new(key, message.encode('utf-8'), hash\_function).digest()

    return mac

def verify\_mac(key, message, received\_mac):

    # Use SHA-256 as the hash function, you can choose a different one if needed

    hash\_function = hashlib.sha256

    # Generate a new MAC using the provided key and message

    generated\_mac = hmac.new(key, message.encode('utf-8'), hash\_function).digest()

    # Compare the generated MAC with the received MAC

    return hmac.compare\_digest(generated\_mac, received\_mac)

# Example usage:

key = b'secret\_key'

message = 'My name is Shridhar '

# Generate MAC

mac = generate\_mac(key, message)

print(f'Generated MAC: {mac.hex()}')

# Verify MAC

is\_valid = verify\_mac(key, message, mac)

print(f'MAC is valid: {is\_valid}')

**Output:-**

Generated MAC: 5e148b47f3c7e1d8c34dcf766dfd99b9b3dfe7f640c6ec10b48b056b141f576c

MAC is valid: True

**If I change the message The MAC will be change**

# Example usage:

key = b'secret\_key'

message = 'good morning'

**Output:-**

Generated MAC: 31375186a2e025465e60a91a5dee2a96d34782fd3b9bc940f9ba27ea166a89a6

MAC is valid: True