Assignment 3

IDATT2503 - Security in programming and cryptography
Fall 2023

Philipp Peron, Francesca Grimaldi, Landon Aune

1 Technical Details

The JPEG Image format can be viewed in its hex representation, and ends with the delimiter FF D9. Modifying any values between the start and end delimiter would affect the image content, but values can be added after the delimiter and still be saved without affecting the image.

This can be used to hide hex characters in a way that is fairly difficult to detect without thorough investigation. Combining this with hashing further increases the security of data.

For this challenge, we decided to have each character individually using MD5 to introduce additional challenge.

2 How to solve

1. For those who are not familiar with JPEG image format and its hex representation, the first thing would be to look up for some information and find that images start with a marker which always contains the hex values “FF D8 FF”, and end with the values “FF D9”
2. Using any hex viewer it is possible to see that other hex values have been added after the delimiter of our image: there is the “encrypted” flag
3. The command `hex meme.jpg >> hex.txt` allows to copy the hex representation in a text file, so that finding the delimiter and copying the other numbers is easier
4. Take the hex numbers where every character of the flag is hashed with a MD5 hash and copy them into Python for example
5. Brute force every character in the flag by trying out hashes for all characters

```python
import string
list_of_characters = string.ascii_letters + string.digits +
string.punctuation
hash_combined = # Put hex string from file here
# For loop that takes 32 characters at a time from the
hash_combined string
for i in range(0, len(hash_combined), 32):
    # Compare hashes of ascii characters to current hash
    for c in list_of_characters:
        c_hash = hashlib.md5(bytes(c, 'utf-8')).hexdigest()
        if c_hash == hash_combined[i:i+32]:
            print(f"{c}: {c_hash}")
            break
```

Figure 1: Brute force example in Python

3 Hints

1. The delimiter for JPEG images is “FF D9”
2. The challenge has MD5 in its name
3. A MD5 hash is 32 characters long in hex