Watermarking based image encryption/decryption system

Team: EvalIoT [Members: Amey Mhadgut (arm994) and Parijat Parimal (pp2206)]
Part of Practical Computer Security Spring 2021

Agenda

1. Problem statement and use case definition
2. Algorithm and threat model
3. Success metrics and results
4. Future Work and Key Considerations

Problem statement and use case definition

- The role of encryption algorithms **ceases to exist** after a user receives and decrypts an image
- Watermarking of images is a technique that protects an image even **after the decryption***
- Idea: Hide image inside another image
- Possible use cases include: Steganography, copy prevention and copyright protection

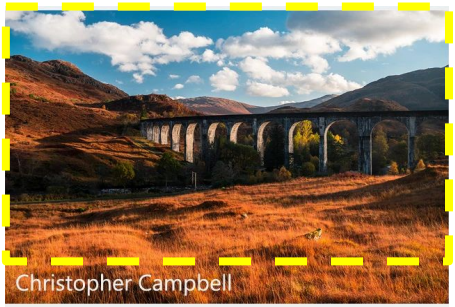


Image A

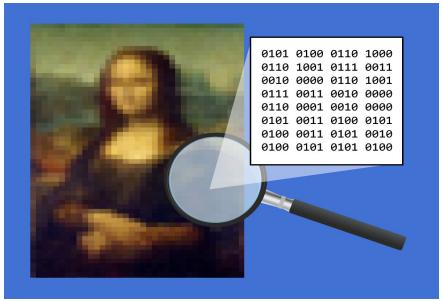
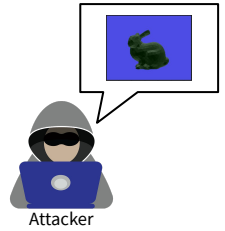


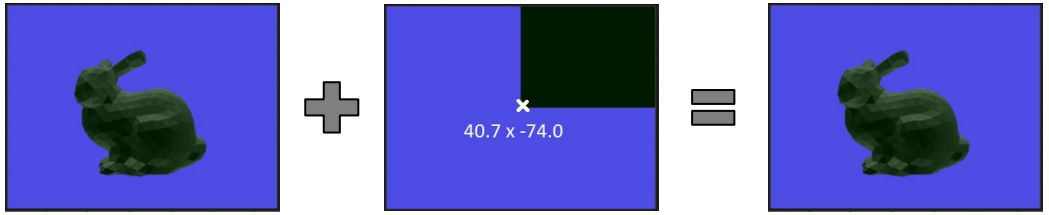
Image B

Image C

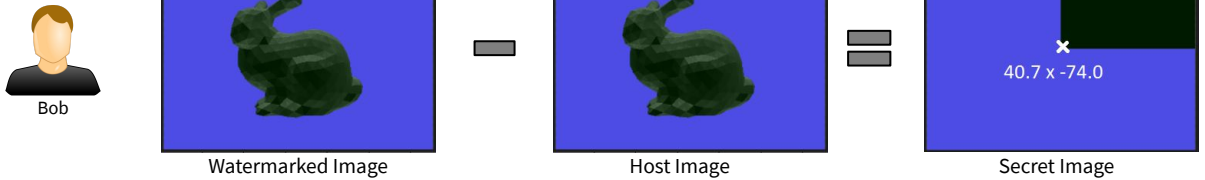
Algorithm & Threat Model

Basic Idea of watermarking



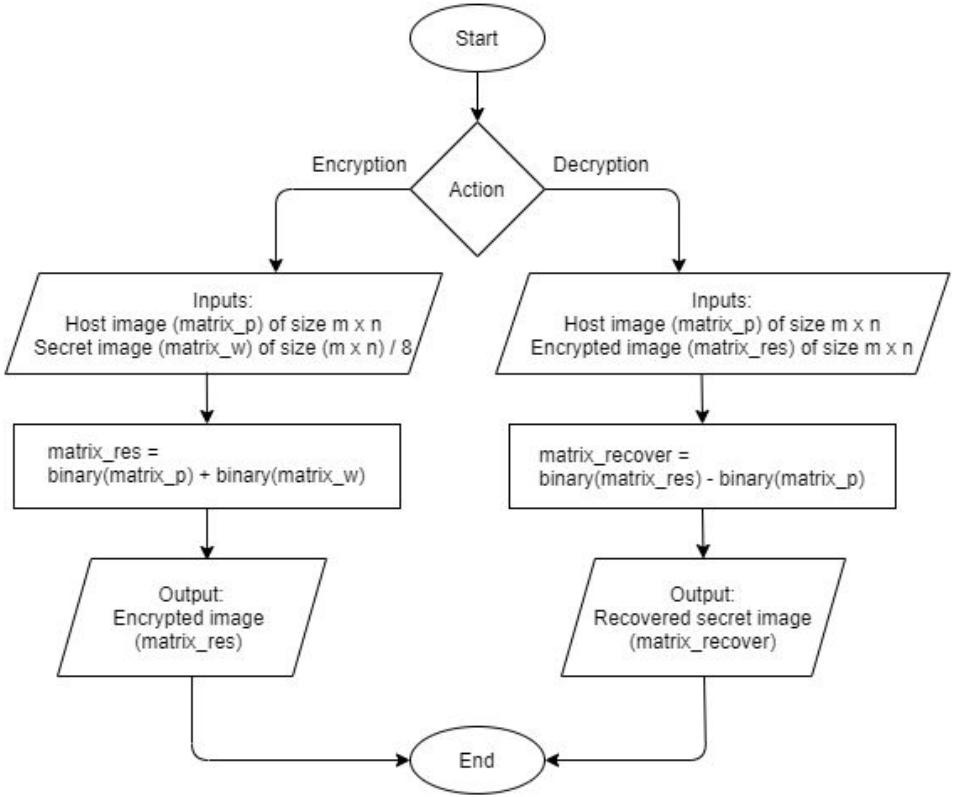


Encrypted



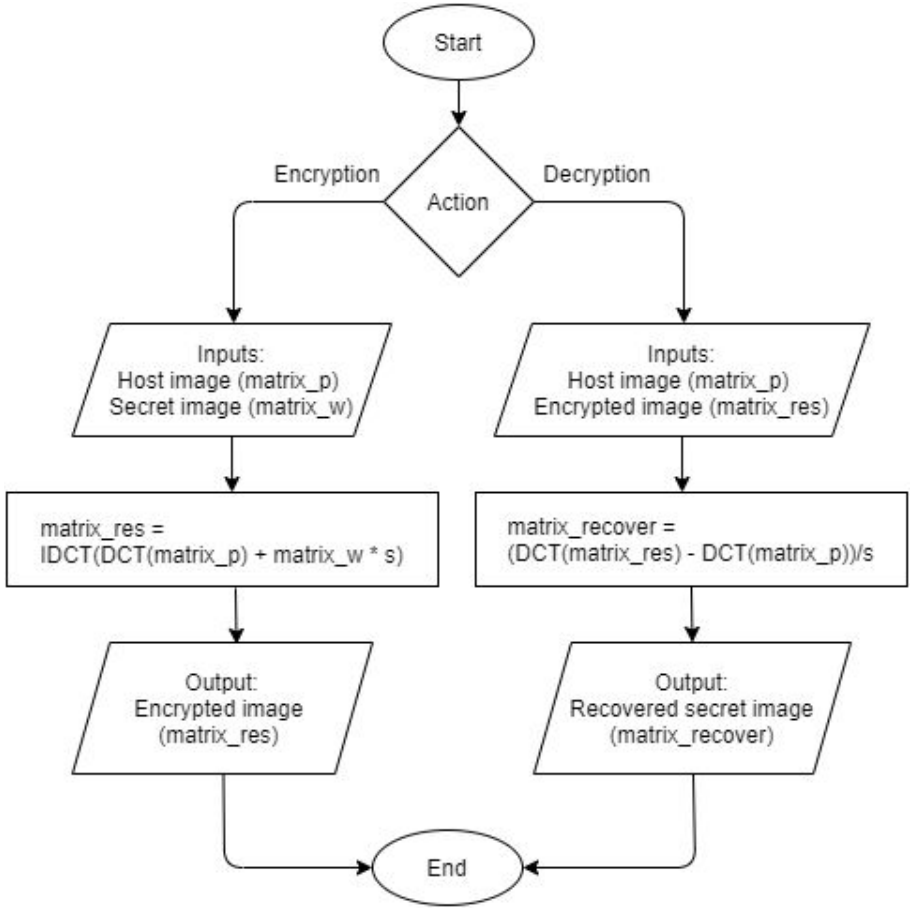
Algorithm #1

Idea: Adding watermark as Least Significant Bit



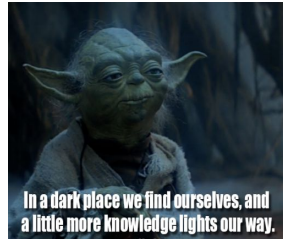
Algorithm #2

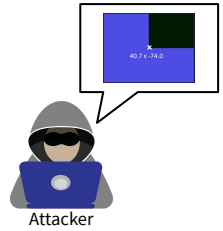
Idea: Using Discrete Cosine Transform with parameter s



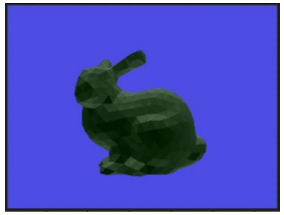
Threat #1

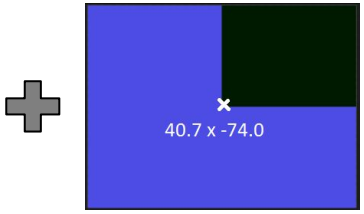
Attacker knows the algorithm used



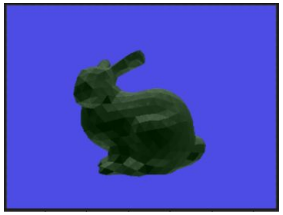



Alice


Host Image

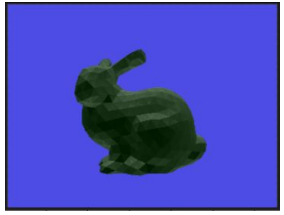


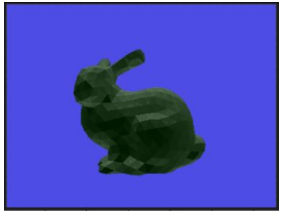
Secret Image

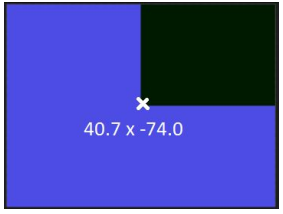

Watermarked Image

Encrypted


Bob


Watermarked Image

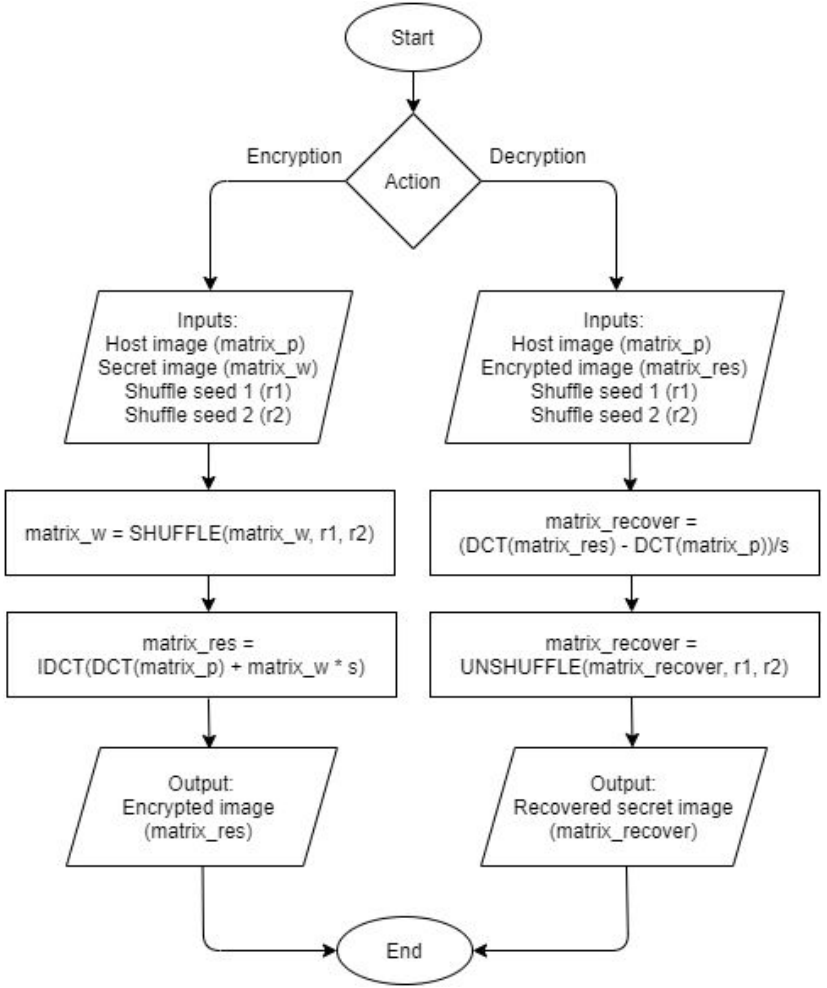

Host Image



Secret Image

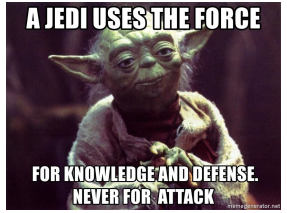
Algorithm #3

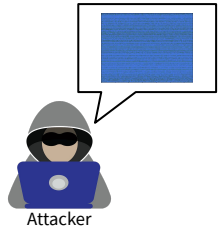
Idea: Using DCT + pixel shuffling based on 2 seed values

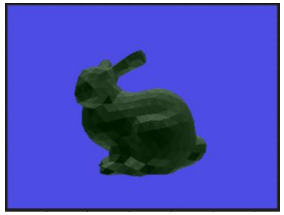


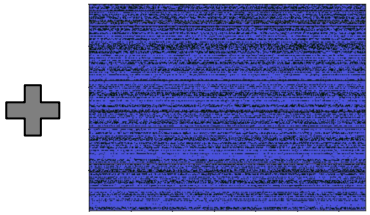
Defense #1

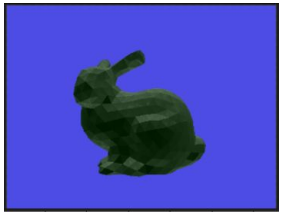
Use 2 seeds R1 & R2 to shuffle the image



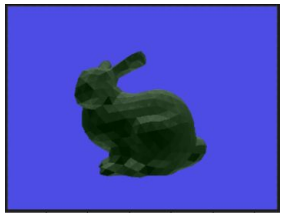


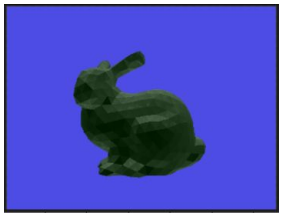

Host Image

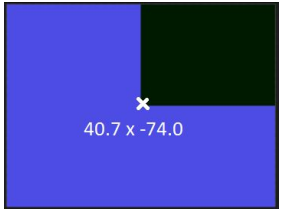

Pixel Shuffled Secret Image


Watermarked Image

Encrypted


Watermarked Image

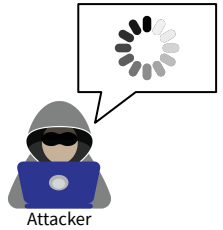

Host Image

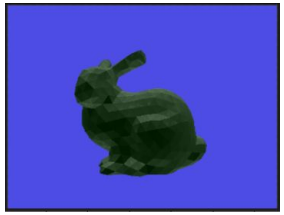


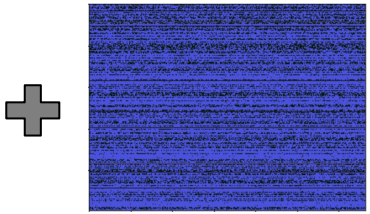
Pixel Unshuffled Secret Image

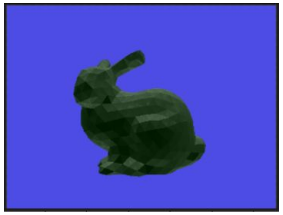
Threat & Defence #2

Attacker tries to brute force the seeds R1 and R2

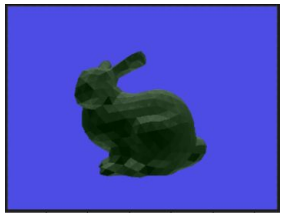


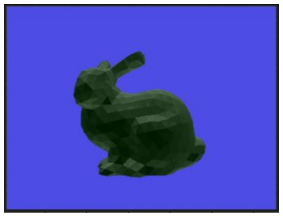

Host Image

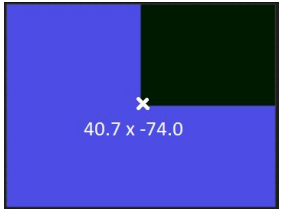

Pixel Shuffled Secret Image


Watermarked Image

Encrypted


Watermarked Image


Host Image



Pixel Unshuffled Secret Image

Success Metrics & Results

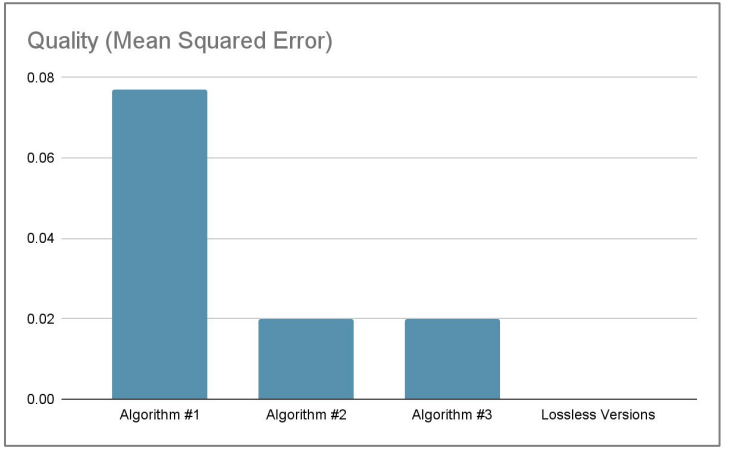
Success metrics

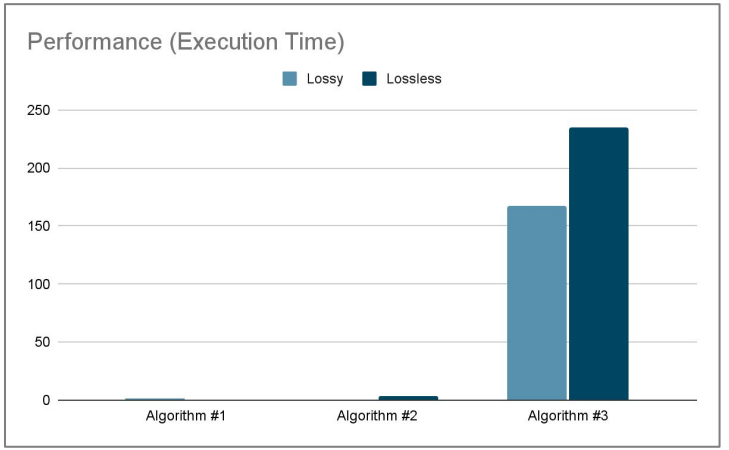
Success metrics can be defined based on two considerations -

1. **Security** against attacks (as discussed)
2. **Quality** of the images: Mean squared error (MSE)
3. **Performance**: Execution time

About loss:

1. Loss was because of floating point to integer conversion
2. We added a mode to get lossless images by saving the floating point values in another format (.txt in our case) and using that for extraction





Key Considerations and Future Work

- Run time
- Brute force attack
- Exploring and integrating with other algorithms

Q&A