# Encryption and Decryption Process Explanation with Example

This document provides a detailed step-by-step explanation of the encryption and decryption processes for a 2x2 image matrix using a key-based algorithm. Each step is illustrated with a mathematical example to make the process clear and understandable.

## Example Setup

For this example, we'll use a 2x2 grayscale image matrix with the following pixel values, and a key of 5:  
Image Matrix (2x2):  
image = [[200, 150],[100, 50]]  
Key = 5

## Encryption Process

Step-by-Step Encryption

1. Load and Flatten the Image

Original Image Matrix:  
 image = [[200, 150], [100, 50]]  
 Flattened Image: [200, 150, 100, 50]

2. Generate PRNG-Based Shuffle Indices

Initialize PRNG with key = 5. Assume PRNG yields indices [3, 0, 2, 1], resulting in:  
 shuffled\_indices = [1, 3, 2, 0]

3. Shuffle Pixels

Applying shuffled\_indices to flat\_image:  
 shuffled\_image = [150, 50, 100, 200]  
 Reshaped back to 2x2 matrix:  
 [[150, 50], [100, 200]]

4. XOR Encryption with Dynamic Key

Dynamic key calculation for each pixel:  
 - (0,0): dynamic\_key = 5; 150 XOR 5 = 155  
 - (0,1): dynamic\_key = 6; 50 XOR 6 = 52  
 - (1,0): dynamic\_key = 6; 100 XOR 6 = 98  
 - (1,1): dynamic\_key = 7; 200 XOR 7 = 207  
 Resulting encrypted image: [[155, 52], [98, 207]]

5. Shift Rows and Columns

Shift by 1 row and 1 column (key % cols and key % rows):  
 Final encrypted image: [[207, 98], [52, 155]]

## Decryption Process

Step-by-Step Decryption

1. Reverse Row and Column Shifts

Reverse the shifts:  
 [[207, 98], [52, 155]] → [[155, 52], [98, 207]]

2. XOR Decryption with Dynamic Key

Use same dynamic keys:  
 - (0,0): 155 XOR 5 = 150  
 - (0,1): 52 XOR 6 = 50  
 - (1,0): 98 XOR 6 = 100  
 - (1,1): 207 XOR 7 = 200  
 Partially decrypted image: [[150, 50], [100, 200]]

3. Reverse Shuffle

Using shuffled\_indices to reverse the shuffle order:  
 Reversed to original flattened form: [200, 150, 100, 50]  
 Reshape to 2x2 matrix:  
 [[200, 150], [100, 50]]

## Conclusion

Following these encryption and decryption steps, the original image matrix is successfully reconstructed. This example demonstrates how pixel shuffling, XOR encryption, and shifts are applied and reversed to achieve secure encryption and decryption using a key.