# Cryptographic methods Using matrix operations

Dinesh Nariani, Alen Kuriakose, Riddhi Parekh,

Abhiram Srikanth, Samasti Bhatt

# QUESTION WE WANT TO ADDRESS

> What is cryptography ?

> Our motivation behind choosing this subject

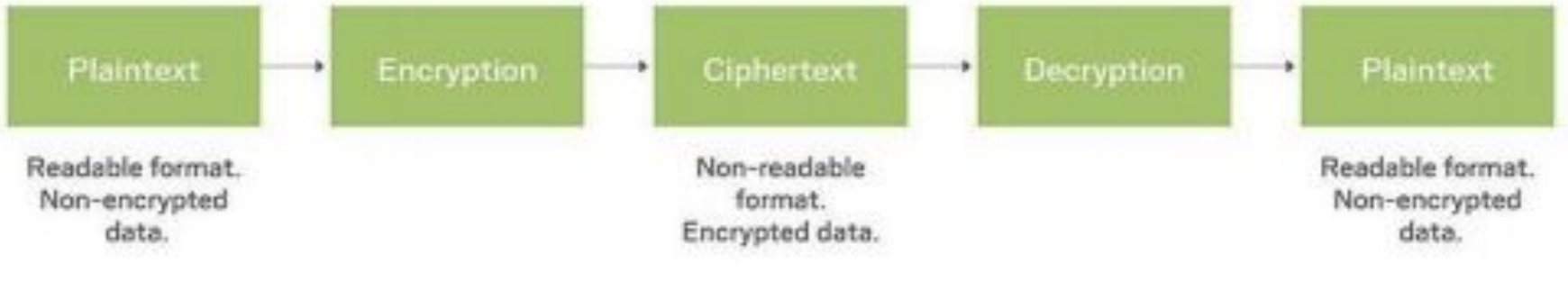
> Aim of cryptography

> How we can use linear algebra (matrices) to encrypt & decrypt image and text

# What is cryptography?

> Cryptography is basically a method used to secure and protect data in any kind of communication.

> It is a process in which plain text is converted into an unreadable format and vice-versa, using encryption and decryption so that only a designated viewer can see it.



# Encryption and decryption

> **Encryption** is a process in which original data is transformed into an unrecognizable form (cipher text)

> **Decryption** converts that encrypted data into a readable form for a human or a computer.

> We can encrypt or decrypt almost any kind of data file or image file.

> For Example,

# Motivation

> Nowadays, authenticity is much more important of an issue.

> In today's world where everything is shifting to digital and at the same time number of increasing cyber attacks and frods, it has become very important to make sure that whatever information we are sharing on any website or via any online platforms or messenger is protected and does not go in the hands of any wrong person.

> We do have passwords as a primary security option but they are somehow not enough as a sole purpose of protection.

> That is why we think that the usage of Cryptography may help us a lot when it comes to authentication.

# AIM OF CRYPTOGRAPHY

There are five primary functions of cryptography ( Read from report in detail )

> Privacy

> Non - repudiation

> Integrity/ Reliability

> Authentication

> Key - exchange

# About Keys

> The process of encryption and decryption involve an **important task of generating keys** . This is where we use the concept of Linear Algebra, our prime topic of concern.

> Key can be a word, phrase or number . Here, since we have used **linear algebraic approach, the key is a matrix**.

> The **key matrix** is used to encrypt the messages, and its inverse is used to decrypt the encoded messages. It is important that the **key matrix** be kept secret between the message senders and intended recipients. If the **key matrix** or its inverse is discovered, then all intercepted messages **can** be easily decoded.

# Text Cryptography - Encryption

> The text required for cryptography is converted to matrix, then transformed to another one using a key matrix and sent to the receiver who receives it and using the key decrypts the information. The step-by-step algorithms are given below.

***Encryption:***

> Initially, a hash table is made. This consists of numbering characters in a text randomly. This can also be done for all the alphabets in a language and then using the required ones from those. It is made sure that the numbers used are unique and not repeated.

> The text is then broken down into multiple parts. These parts have fixed length and is essential to be so here. Let us consider it to be divided into ‘n’ parts.

# Text Cryptography - Encryption

> After this, an approval is required from senders and receivers' part. On approval, a matrix is generated. This is the **key matrix (K)** and is an **invertible matrix**. The dimension of matrix will be **n x n.**

> This is followed by the conversion of each of the n parts to the associated numbers we choose for making the hash table. Each of the parts will form an n x 1 matrix.

> Next, these vectors are transformed. The process uses the key matrix generated in step (c). Congruence modulo method is then used to find total number of characters.

$$T_a : V1 \rightarrow V2$$
$$T_a \equiv K \times \mathbf{mod}\ n$$

> This generates a set of numbers (non-negative always) which are then transformed back to normal characters using the same hash table.

> *The text now in hand is the encrypted text or cipher text.* This text is then transmitted to the receiver in whichever electronic medium preferred.

# Text Cryptography - Decryption

***Decryption :***

> The receiver receives the information – the cipher text. The key and hash table are already known to the receiver. Thus, the process of decryption starts at the same moment the information is received.

> The encrypted text is then divided into ‘n’ part of n x 1 vectors. The same hash table is used for this process.

> After this, the inverse of the key matrix is found (K-1).

> Then, using congruence modulo method and the inverse key, the original vectors of the matrix are found.

$$T_b : V2 \rightarrow V1$$
$$T_b \equiv K\text{-}1 \bmod n$$

> Finally, the hash table is used once again to convert the numbers back to original text, i.e., the information which was sent to the receiver. This is the last step involved in textual decryption and the process is complete.

# Text cryptography Example

>Consider the text " **CONFIDENT"** which is to be encrypted.

>First, we need to create a hash table - a hash table consists of 26 uppercase alphabets with underscore as indicator of space. Hence we will have matric modulo = 27

| A | B | C | D | E | F | G | H | I |
|---|---|---|---|---|---|---|---|---|
| 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 |
| J | K | L | M | N | O | P | Q | R |
| 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 |
| S | T | U | V | W | X | Y | Z | _ |
| 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |

> Now we divide "confident" into multiple parts of length and encode them into vectors of 3x1

$$CON = \begin{bmatrix} 24 \\ 12 \\ 13 \end{bmatrix}_{3\times1} \quad FID = \begin{bmatrix} 21 \\ 18 \\ 23 \end{bmatrix}_{3\times1} \quad ENT = \begin{bmatrix} 22 \\ 13 \\ 7 \end{bmatrix}_{3\times1}$$

> A key matrix is made that is chosen after the approval of both sender and receiver $M = \begin{bmatrix} 1 & 0 & 5 \\ 2 & 1 & 6 \\ 3 & 4 & 0 \end{bmatrix}_{3\times3}$

> By Gauss-Jordan Method inverse of key matrix is $M^{-1} = \begin{bmatrix} -24 & 20 & -5 \\ 18 & -15 & 4 \\ 5 & -4 & 1 \end{bmatrix}_{3\times3} * (mod\ 27)$

$$M^{-1} = \begin{bmatrix} 3 & 20 & 22 \\ 18 & 12 & 4 \\ 5 & 23 & 1 \end{bmatrix}_{3\times3}$$

> Now the vectors are linearly transformed to another set of vectors called cipher vectors.

$$\begin{bmatrix} 1 & 0 & 5 \\ 2 & 1 & 6 \\ 3 & 4 & 0 \end{bmatrix}_{3\times3} * \begin{bmatrix} 24 \\ 12 \\ 13 \end{bmatrix}_{3\times1} = \begin{bmatrix} 89 \\ 138 \\ 120 \end{bmatrix}_{3\times1} * (mod\ 27) = \begin{bmatrix} 8 \\ 3 \\ 12 \end{bmatrix}_{3\times1}$$

$$\begin{bmatrix} 1 & 0 & 5 \\ 2 & 1 & 6 \\ 3 & 4 & 0 \end{bmatrix}_{3\times3} * \begin{bmatrix} 21 \\ 18 \\ 23 \end{bmatrix}_{3\times1} = \begin{bmatrix} 136 \\ 198 \\ 135 \end{bmatrix}_{3\times1} * (mod\ 27) = \begin{bmatrix} 1 \\ 9 \\ 0 \end{bmatrix}_{3\times1}$$

$$\begin{bmatrix} 1 & 0 & 5 \\ 2 & 1 & 6 \\ 3 & 4 & 0 \end{bmatrix}_{3\times3} * \begin{bmatrix} 22 \\ 13 \\ 7 \end{bmatrix}_{3\times1} = \begin{bmatrix} 57 \\ 99 \\ 135 \end{bmatrix}_{3\times1} * (mod\ 27) = \begin{bmatrix} 3 \\ 18 \\ 10 \end{bmatrix}_{3\times1}$$

**Encrypted matix**

> By using hash table, $\begin{bmatrix}3\\18\\10\end{bmatrix}_{3\times1} = XIQ \quad \begin{bmatrix}8\\3\\12\end{bmatrix}_{3\times1} = SXO \quad \begin{bmatrix}1\\9\\0\end{bmatrix}_{3\times1} = ZR\_\_$

**Hash code**

| A | B | C | D | E | F | G | H | I |
|---|---|---|---|---|---|---|---|---|
| 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 |
| J | K | L | M | N | O | P | Q | R |
| 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 |
| S | T | U | V | W | X | Y | Z | __ |
| 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |

> Now, Transforming the matrices using $M^{-1}$ gives the original message.

$$\begin{bmatrix}3&20&22\\18&12&4\\5&23&1\end{bmatrix}_{3\times3} * \begin{bmatrix}8\\3\\12\end{bmatrix}_{3\times1} = \begin{bmatrix}348\\228\\121\end{bmatrix}_{3\times1} * (mod\ 27) = \begin{bmatrix}24\\12\\13\end{bmatrix}_{3\times1}$$

**Decrypted matrix**

$$\begin{bmatrix}24\\12\\13\end{bmatrix}_{3\times1} = CON \quad \begin{bmatrix}21\\18\\23\end{bmatrix}_{3\times1} = FID \quad \begin{bmatrix}22\\13\\7\end{bmatrix}_{3\times1} = ENT$$

$$\begin{bmatrix}3&20&22\\18&12&4\\5&23&1\end{bmatrix}_{3\times3} * \begin{bmatrix}1\\9\\0\end{bmatrix}_{3\times1} = \begin{bmatrix}183\\126\\212\end{bmatrix}_{3\times1} * (mod\ 27)$$

$$= \begin{bmatrix}21\\18\\23\end{bmatrix}_{3\times1}$$

$$\begin{bmatrix}3&20&22\\18&12&4\\5&23&1\end{bmatrix}_{3\times3} * \begin{bmatrix}3\\18\\10\end{bmatrix}_{3\times1} = \begin{bmatrix}589\\310\\439\end{bmatrix}_{3\times1} * (mod\ 27)$$

$$= \begin{bmatrix}22\\13\\7\end{bmatrix}_{3\times1}$$

# Image - Encryption

> Here we use the RGB pixel data and also transform size of matrices. The three-dimensional vector space is transformed to a two-dimensional RGB image. As previously key generations are involved and are made sure they are unique and no patterns exist which makes them exploitable.

***Encryption:***

> The image to be encrypted is chosen and converted to two-dimensional RGB pixel data. This makes it into a form which is easier to use and manipulate.

> Then, the image is transformed and represented into a two-dimensional matrix (P) of the size m x n. Linear transformations of matrix is the exact process which happens in this step.

> After this, a random m x m matrix is generated. This acts as the key matrix (K) for the process. The P matrix is then multiplied with the K matrix to get the encrypted matrix.

> The encrypted matrix is then transformed back to the three-dimensional matrix with encrypted RGB pixel data.

> This is then sent to the receiver.

# Image - Decryption

***Decryption :***

> The receiver receives the information – the encrypted 3D image.

> The matrix is transformed to the two-dimensional version. This is followed by the key verification for the two-dimensional matrix.

> The inverse key is found (K-1)

> The inverse key is then multiplied with the two-dimensional matrix to get the original two-dimensional decrypted matrix P.

> Finally, matrix P, which is two-dimensional is converted back to the original three-dimensional image with RGB data using the same conversion techniques. This is the last step involved and we get the original image and the process is complete.

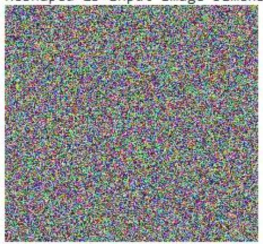
# Results

Input Image Dimension = 225

Key Dimension = (675, 675)

***Original Image***

Reshaped 2D Input Image Dimension = (675, 225)



Decrypted 2D Image Dimension = (675, 225)

***Encrypted Image***

Decrypted Image Dimension = (225, 225, 3)

Time : 0.3425

***Decrypted Image***

# APPLICATIONS

> Authentication and Digital Signatures

> Time Stamping

> Electronic Money

> Secure Network Communications

- Secure Socket Layer (SSL)
- Kerberos

> Anonymous Remailers

> Disk Encryption

> Quantum Cryptography

# Thank you