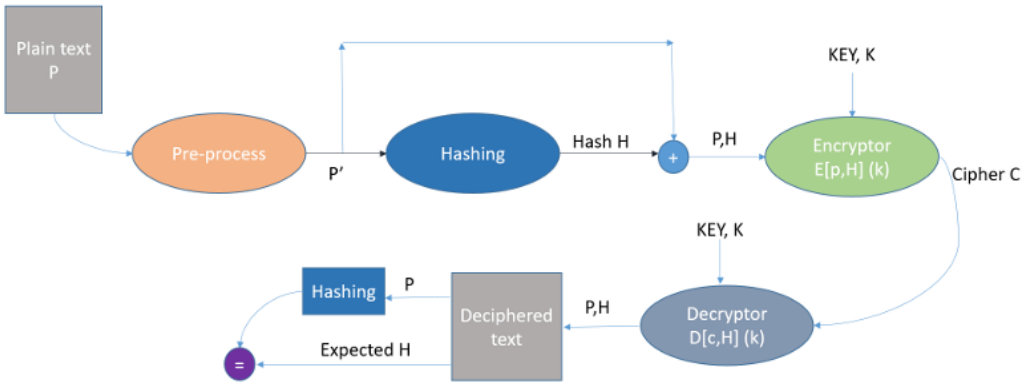
Project1: Encryption and Decryption Using Transposition Cipher



Description of system

1. Preprocessing

- Input Sanitization:
  Filters the plaintext to include only lowercase letters a-z.
- Padding:
  Appends random characters to ensure the total length (plaintext + hash) is divisible by the key length.
- Goal:
  Ensures all rows are fully filled during matrix-based encryption.

2. Hash Function

- Input Format:
  Accepts a preprocessed string s = $s_1s_2s_3...s_n$.
- Matrix Fill:
  Fills the input into a matrix with 8 columns.
- Hash Initialization:
  Starts with a hash code of 8 'a' characters (i.e., zeroed values).
- XOR & Rotate:
  Each character is XOR-ed with corresponding hash column values.
  Left rotation of the hash array is performed after processing each row.

3. Encryptor (Transposition Cipher)

- Matrix Form:
  Converts the plaintext into a matrix using the key length as the number of columns.
- Column Permutation:
  Reorders columns based on the key vector.
- Ciphertext Extraction:
  Reads column-by-column to generate the ciphertext.

4. Decryptor

- Matrix Form:
  Reconstructs the matrix from ciphertext using the key length.
- Column Placement:
  Columns are placed according to the inverse permutation of the key.
- Plaintext Reconstruction:
  Reads row-by-row to retrieve the original padded+hashed plaintext.

5. Property Check (π)

- Purpose:
  Validates decryption success without prior knowledge of the original plaintext.
- How It Works:
  The last 8 characters of the plaintext are expected to be a hash of the preceding characters.
  π holds true if:
  hash(plaintext[:-8]) == plaintext[-8:]

6. Brute Force Attack

- Key Range:
  Tries all permutations of key lengths from 2 to 9.
- Phase 1 – Candidate Discovery:
  Applies brute-force decryption on one ciphertext sample using all possible keys.
  Filters keys satisfying the Property Check π.
- Phase 2 – Global Validation:
  Applies each candidate key to the remaining 4 ciphertexts.
  Correct key is the one that satisfies all π conditions.

Expected output

Randomly Generated Key: [4, 8, 5, 7, 2, 6, 1, 3, 9]

Sample-1----------

-> plainText: Hi I'm Mangala Manmatharaja

-> processedText: hiimmangalamanmatharajalwtfq

-> hashvalue: dtuqdhdq

-> hashedText: hiimmangalamanmatharajalwtfqdtuqdhdq

-> cipherText: nawhmnaqgttdhlaqimatamldmajuiardahfq

-> decipherText: hiimmangalamanmatharajalwtfqdtuqdhdq

Sample-2----------

-> plainText: I'm currently doing my degree program at SEUSL

-> processedText: imcurrentlydoingmydegreeprogramatseuslhpxxuafz

-> hashvalue: qfayzbut

-> hashedText: imcurrentlydoingmydegreeprogramatseuslhpxxuafzqfayzbut

-> cipherText: egpsubrieaxynmreauildgszcdgahfrnetxzuormpamyerlqtyouft

-> decipherText: imcurrentlydoingmydegreeprogramatseuslhpxxuafzqfayzbut

---

Sample-3----------

-> plainText: I'm a Network student

-> processedText: imanetworkstudentzj

-> hashvalue: zgehvkfd

-> hashedText: imanetworkstudentzjzgehvkfd

-> cipherText: wnkedhotfikjatgtevnuemszrzd

-> decipherText: imanetworkstudentzjzgehvkfd

Sample-4----------

-> plainText: I have been assigned project 1

-> processedText: ihavebeenassignedprojectkfrs

-> hashvalue: ccwznozd

-> hashedText: ihavebeenassignedprojectkfrsccwznozd

-> cipherText: eekoegczedfziarsasjcbntnviewhsocnprd

-> decipherText: ihavebeenassignedprojectkfrsccwznozd

Sample-5----------

-> plainText: This project involves transposition cipher system with brute force attack

-> processedText: thisprojectinvolvestranspositionciphersystemwithbruteforceattack

-> hashvalue: fgblmeqw

-> hashedText:
thisprojectinvolvestranspositionciphersystemwithbruteforceattackfgblmeqw

-> cipherText:
olpiettepvnnsraljvopmeaqtcsieiokiirishcgrosctutmsnaoybebhtttrtrfeeshwfcw

-> decipherText:
thisprojectinvolvestranspositionciphersystemwithbruteforceattackfgblmeqw

---

Launching brute force ...

Starting key search from length 1 to 9

Woooowhooh! Found key : (4, 8, 5, 7, 2, 6, 1, 3, 9)