University of Jeddah

College of Computer Science and Engineering Department of Cybersecurity

CCCY 312: Cryptography Project

~N01 Algorithm

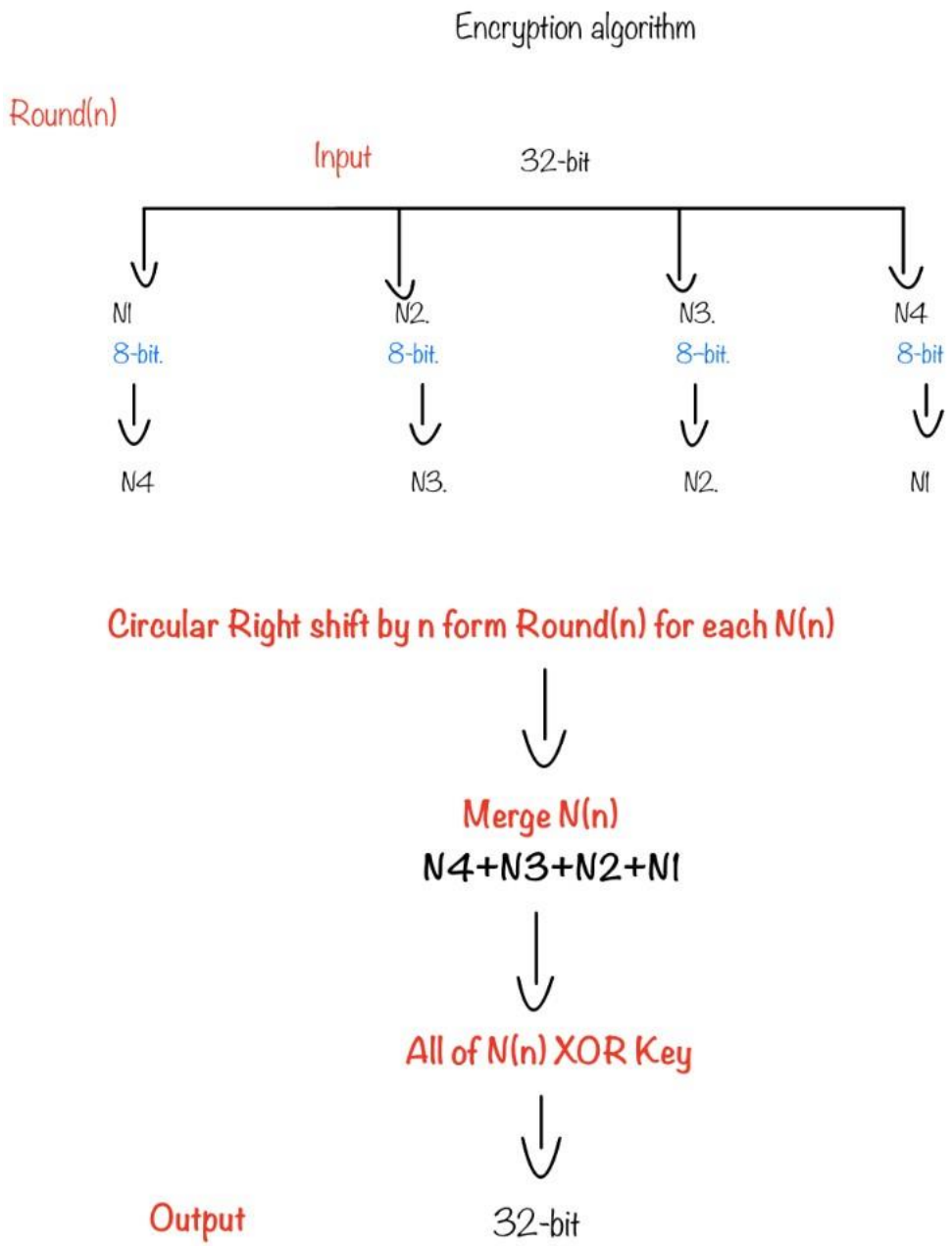
Project members

Abdullah Alhakami - 2041050

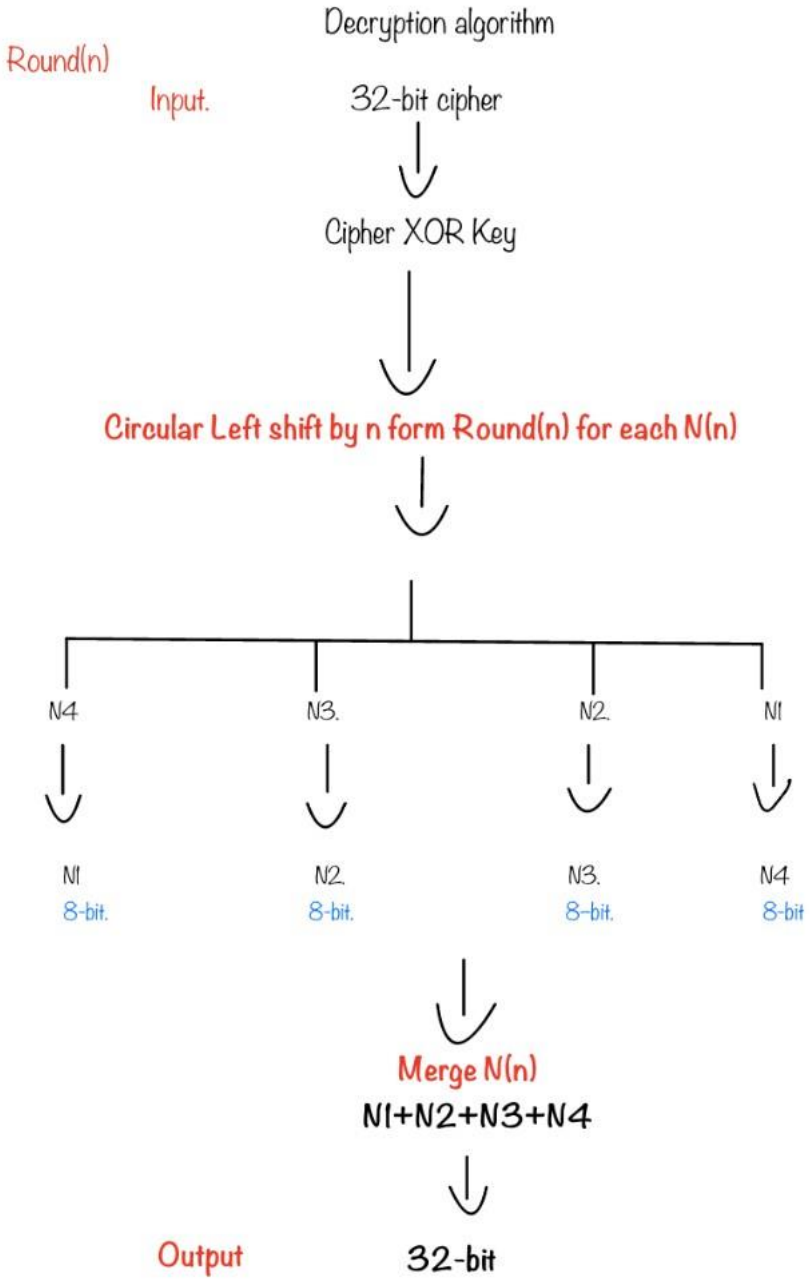
Abdulaziz Alzahrani - 2040847

Assim Felemban - 2042005

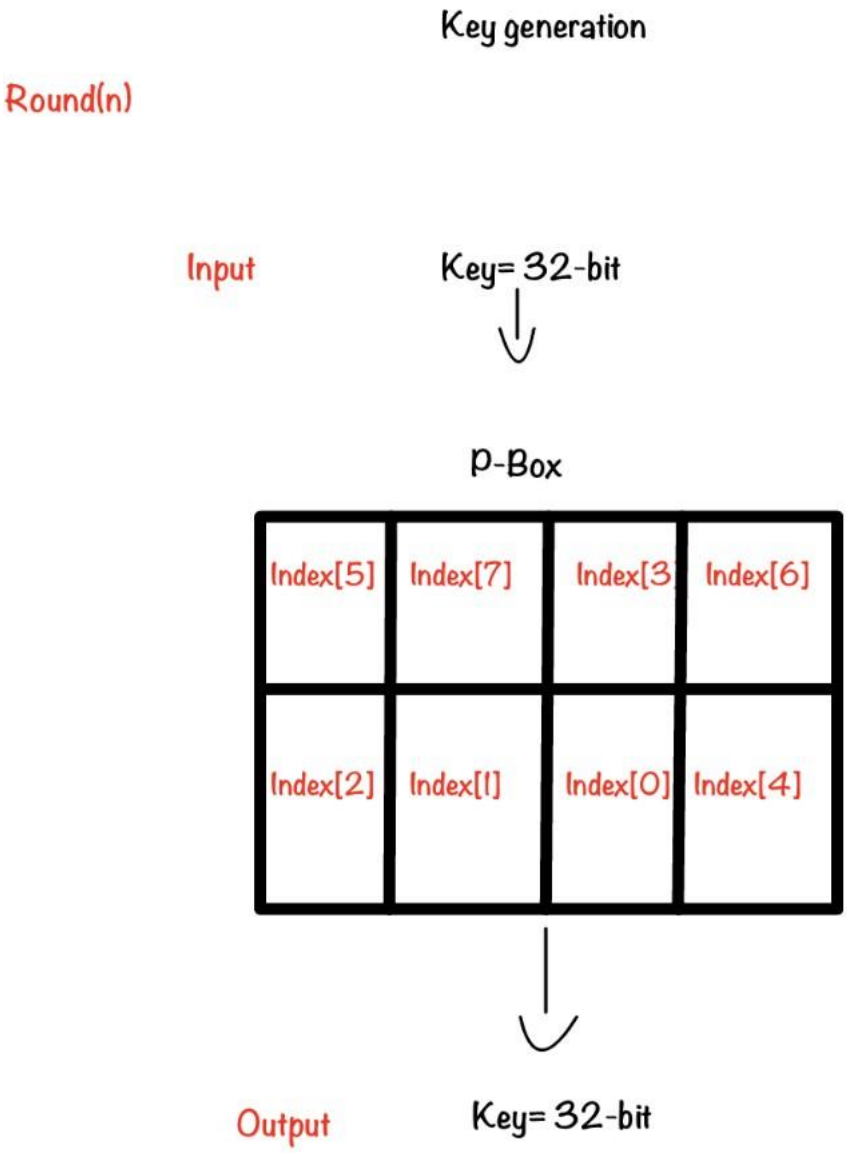
Encryption algorithm: in this algorithm the input will be 32-bit plaintext and, divided into 4 parts N(1..4). After that will perform transposition and do circular Right shift by the round number. N(1..4) will be merged into one N . N will Perform XOR with key. The output 32-bit cipher is the result from XOR operation.



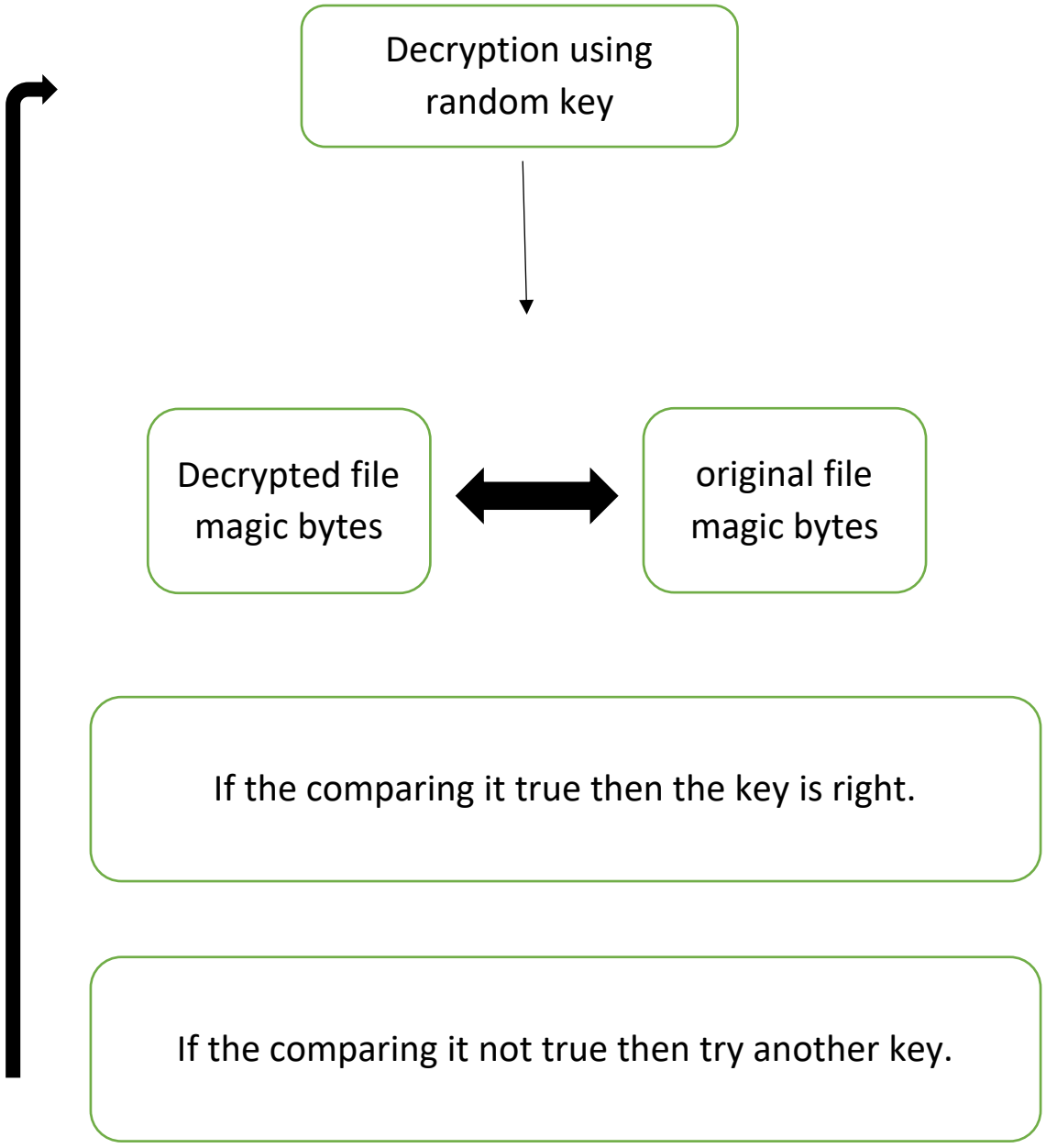
Decryption algorithm: the input is the cipher 32-bit and, will perform XOR with key then do the circular Left shift by the round number. After that divided into 4 parts N (4..1) then will perform transposition operation. N (1..4) merged and, the output is 32-bit plaintext.



Key generation: the key consists of 32 bits. However, the key will go through the P-Box and the order as shown below. The output is 32-bit key(n)



Key brute forcing: by generating a list of all possible 8 hex values like: 029186AB we can brute force the key, by trying to decrypt the encrypted file with every key. We could make sure of the right key by checking the decrypted file magic bytes, for example: if the original file is a pdf file, then we can check the file magic bytes then compare it with the decrypted file, if it has the same values then we found the right key.



Key algorithm weaknesses: by reversing the P-box we used, we can retrieve the original key, doing it 3 times we will retrieve the original key.

P-box

INDEX [0] = INDEX [6]
INDEX [1] = INDEX [5]
INDEX [2] = INDEX [4]
INDEX [3] = INDEX [2]
INDEX [4] = INDEX [7]
INDEX [5] = INDEX [0]
INDEX [6] = INDEX [3]
INDEX [7] = INDEX [1]

Reverse P-box

INDEX [5] = INDEX [1]
INDEX [7] = INDEX [4]
INDEX [3] = INDEX [6]
INDEX [6] = INDEX [0]
INDEX [2] = INDEX [3]
INDEX [1] = INDEX [7]
INDEX [0] = INDEX [5]
INDEX [4] = INDEX [2]