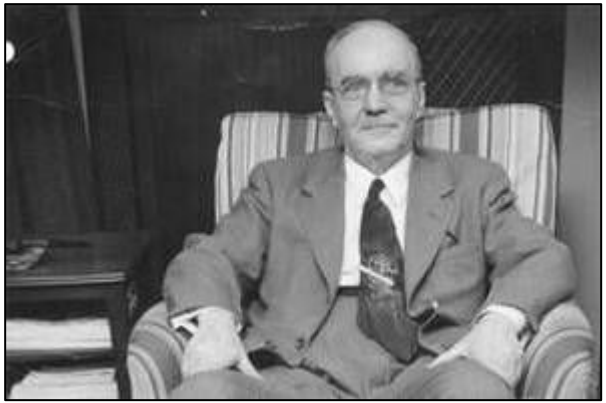
The Hill Cipher

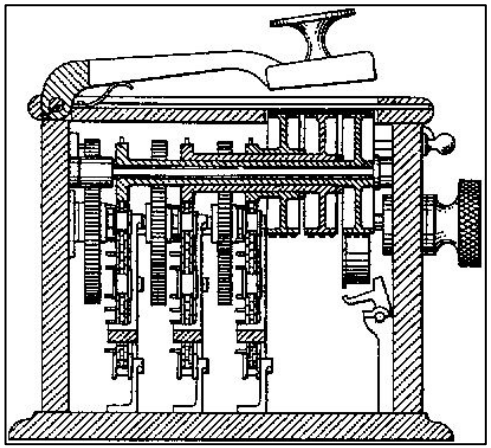
Michael Russo and Jasveenkaur Wahan

HISTORY of the Hill Cipher

Invention



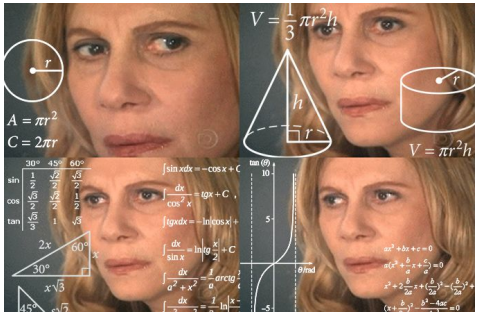
- It was invented by Lester S. Hill in 1929
- The first substitution cipher to allow operations on more than 3 characters at a time
- The cipher uses matrices and linear algebra (we'll explain this soon!)



- Lester Hill got a patent for a mechanical implementation of a Hill cipher of dimension 6 (6 x 6 matrix), but wasn't very secure because its key was fixed

How It Works

Math time

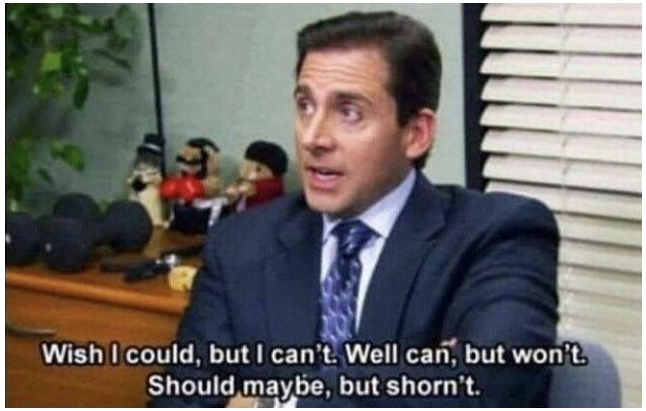


The process of the cipher is surprisingly simple! To encrypt, you just need a few different steps:

1. Get a key (and turn it into a matrix if it isn't already one)
    - We used a word as a key and turned each letter into its numerical equivalent (a = 0, b = 1, etc.)
2. Turn the message you want to encrypt into a series of *digraphs*.
    - A word like "alpha" would be turned into:

      $\begin{pmatrix} a \\ l \end{pmatrix} \begin{pmatrix} p \\ h \end{pmatrix} \begin{pmatrix} a \\ z \end{pmatrix}$ a z was added at the end of that in order to complete the digraph!
3. Multiply your key matrix by each digraph, and then take that resulting matrix mod 26.
4. Turn each number back into its letter form

Decryption



The steps for decryption are also surprisingly simple!

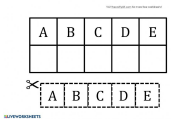
1. Find the inverse matrix of the key.
2. Follow all the steps for encryption, except use the inverse key instead of the regular key and use the encrypted message.
3. Your final result should be a decrypted message! It may have the letter “z” at the end if the message originally had an odd number of characters.

Advantages (and Disadvantages)

Advantages

- The security of permutations of the cipher scale with size; the standard 2x2 hill cipher can be relatively easy to break, but variations with more dimensions theoretically become exponentially less so.
    - They also becomes a lot more difficult to code.

- In order to decrypt the cipher without the key, you must know at least four distinct letters and what they match up to in the encoded text, then augment the matrices of each of these four letter segments. That process seems to be impossible to automate, or at least extremely difficult since it requires logic, and would become very convoluted with larger matrices.



- There is not a 1 to 1 correspondence of letters in the plaintext and the ciphertext, so individual letter frequency analysis doesn't really work on the cipher. Furthermore, n-gram frequency analysis would require a long message in order to work, so the hill cipher is fairly secure in this regard.

Disadvantages

- If some of the content of your encrypted message was known, it would be very easy to obtain the key.
- Expanding on the former, the cipher is susceptible to a known plaintext attack, making n-gram frequency analysis a threat, so anyone with time and expertise would be able to crack the code (but that's true for most ciphers).
  - The required amount of time would decrease with the length of the message
- The cipher itself can only encrypt plaintext blocks to ciphertext blocks, so if its implemented with an image, then it has some issues; large areas of a single color cannot be encrypted and not all features of encrypted images are hidden.

Vulnerability Exploitation

- If you know four distinct characters, you can match them with their numerical values as shown, then convert them into vectors (c_1, c_2 for the ciphertext and p_1, p_2 for the plaintext):

Table A – Letters and Their Corresponding Positions

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

Ciphertext			
J	Y	Q	O
10	25	17	15

Plaintext			
A	T	O	M
1	20	15	13

Ciphertext

$$J \rightarrow \begin{bmatrix} 10 \\ 25 \end{bmatrix}$$
$$Y \rightarrow \begin{bmatrix} 25 \\ 17 \end{bmatrix}$$
$$Q \rightarrow \begin{bmatrix} 17 \\ 15 \end{bmatrix}$$
$$O \rightarrow \begin{bmatrix} 15 \\ 13 \end{bmatrix}$$

Plaintext

$$A \rightarrow \begin{bmatrix} 1 \\ 20 \end{bmatrix}$$
$$T \rightarrow \begin{bmatrix} 20 \\ 15 \end{bmatrix}$$
$$O \rightarrow \begin{bmatrix} 15 \\ 13 \end{bmatrix}$$
$$M \rightarrow \begin{bmatrix} 13 \\ 1 \end{bmatrix}$$

- Transpose the aforementioned vectors and create 2x2 vectors as shown:

$$c_1^T = \begin{bmatrix} 10 & 25 \end{bmatrix}$$
$$c_2^T = \begin{bmatrix} 17 & 15 \end{bmatrix}$$

$$p_1^T = \begin{bmatrix} 1 & 20 \end{bmatrix}$$
$$p_2^T = \begin{bmatrix} 15 & 13 \end{bmatrix}$$

$$C = \begin{bmatrix} c_1^T \\ c_2^T \end{bmatrix} = \begin{bmatrix} 10 & 25 \\ 17 & 15 \end{bmatrix}$$

$$P = \begin{bmatrix} p_1^T \\ p_2^T \end{bmatrix} = \begin{bmatrix} 1 & 20 \\ 15 & 13 \end{bmatrix}$$

- Then, augment the matrices such that [C | P]

- You perform elementary row operations on the matrix [C | P] until the identity matrix is obtained on the right side, to achieve the final product…
  - The key matrix is on the right side
- With the key, you can decode as you normally would!

$$\left[\begin{array}{cc|cc} 1 & 0 & 24 & 19 \\ 0 & 1 & 5 & 14 \end{array}\right]$$

- If you didn't know four distinct matching letters, you could use n-gram frequency analysis to find potential combinations, which is something that we've discussed in class.


ACCORDING TO SOURCES
makeameme.org

Source List

- https://www.cs.jhu.edu/~cgarman/Cryptography.html
- https://mazharhussainatisp.files.wordpress.com/2016/04/lecture-13-network-security.pdf
- https://crypto.interactive-maths.com/hill-cipher.html
- https://link.springer.com/article/10.1631/jzus.2006.A2022
- http://www.math.utah.edu/~gustafso/s2017/2270/projects-2017/saleemaQaziJuliaVonessenGrabrieleLegaspi/Hill%20Ciphers%20Semester%20Project.docx