**Batch: B1 Roll No.: 16010121045**

**Experiment No. 1**

|  |
| --- |
| **Title:**  Cryptanalysis Tools |

**Objective:**

**Expected Outcome of Experiment: To implement Cryptanalysis Tools .**

|  |  |
| --- | --- |
| **CO** | **Outcome** |
|  |  |

**Books/ Journals/ Websites referred:**

[**https://resources.infosecinstitute.com/topics/cryptography/cryptanalysis-tools/**](https://resources.infosecinstitute.com/topics/cryptography/cryptanalysis-tools/)

**Abstract**:-

**Some cryptanalysis tools**

**Brute force attack**

As an old technique, brute force means exhausting very possibility until a match is found. Even in classic cryptography, brute force attack is considered time-consuming.

In modern cryptography, the length of a brute force attack depends exponentially on the length of the key. Since modern cryptography uses very long keys, brute force attack is considered inefficient for all practical purposes.

**Chosen plaintext attack**

The attacker, in this case, inputs a plaintext and observes the output ciphertext obtained. By examining the plaintext – ciphertext pair, he can easily guess the encryption key. The differential analysis done on RSA algorithm is an example of such attack.

**Man in the middle attack**

In this type of attack, Eve fools both Alice and Bob. Alice, who wants to communicate with Bob, relays her public key Ka. Eve impersonates Bob and sends her public key Ke. Alice transmits her plaintext P alongwithKa&Ke.

Now, Eve has Alice's key as well as the plaintext. She now impersonates Alice and sends her key as Alice's key to Bob. Bob transmits his public key Kb to Eve. To keep Bob from suspecting anything, Eve transmits P along with Kb &Ke to Bob.

Now, Eve has both the public keys of Alice and Bob, as well as the message i.e. the real information she needed.

**Tools : -**

**CrypTool**

CrypTool was first launched in 1998. It is an e – learning tool explaining cryptanalysis and cryptography. CrypTool aims at making people understand network security threats and working of cryptology. It includes asymmetric ciphers like RSA, elliptic curve cryptography. CrypTool1 (CT1) experiments with different algorithms and runs on Windows. It was developed in C++ language.

CT2, which was launched in 2014, also runs on Windows. It has an improved GUI and more than hundred cryptological functions. It is developed in .NET &C#. JCrypTool (JCT) which followed CT2 is platform independent.

JCT works on Linux, MacOS, and Windows. JCT is both a function – centric as well as a document – centric tool.In 2009, CrypTool – online (CTO) was launched.CTOconsists of a huge number of encryption methods and analysis tools. It is a web browser based tool and also targeted at smartphones.

****

Here we can crack AES either alone on your computer, or distributed as a team of many clients who connect their browsers ad-hoc over the internet (assuming, the key is partly known).

**The Working**

Enter the ciphertext in hexadecimal encoding as full 128-bit blocks (the preset ciphertext consists of 640 hex characters = 2560 bit = 20 full AES blocks).

Select key length: 128, 192, or 256 bits.

Enter the key in hex:

Enter known digits as hex chars (A-F, 0-9).

Mark unknown hex digits wit a star (\*).

One hex digit correlates with 4 bit. 128 bit correspond to 32 hex digits.

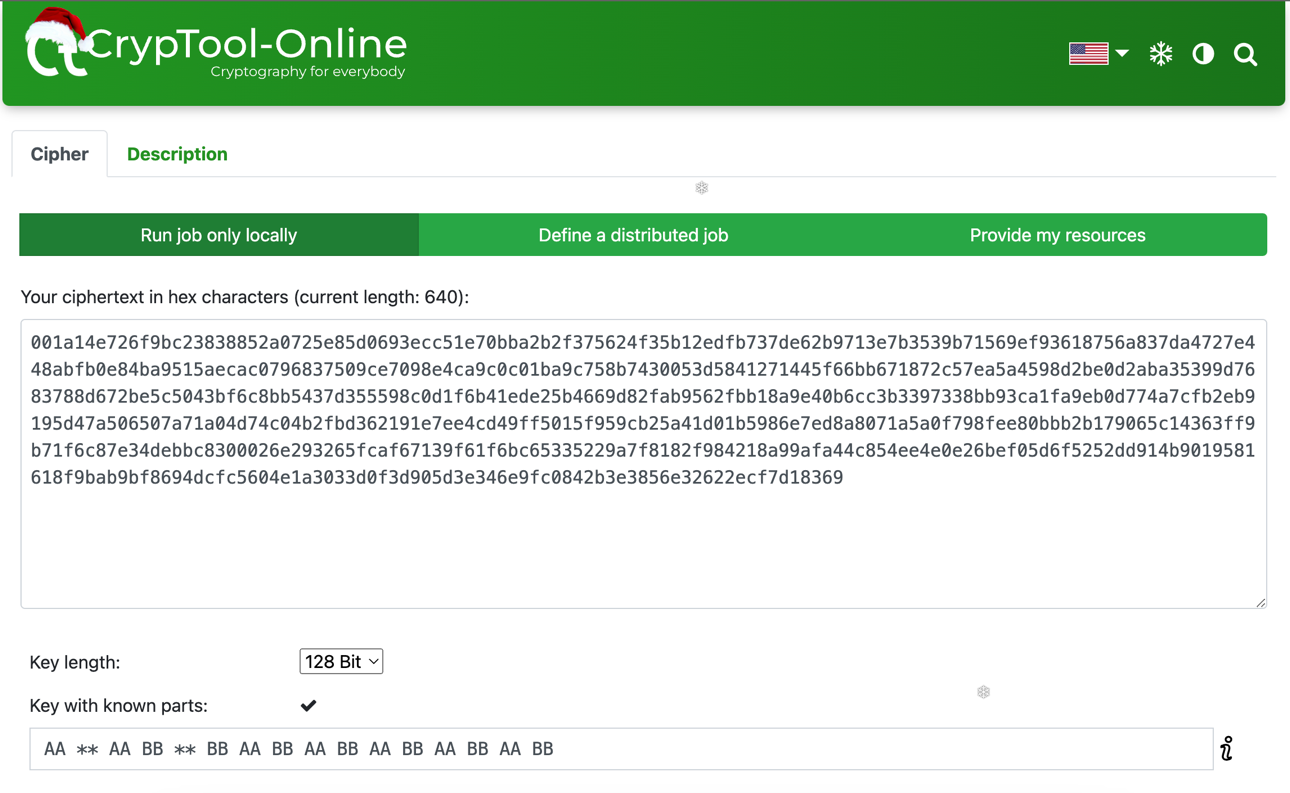
The more \* are specified, the longer the analysis takes.

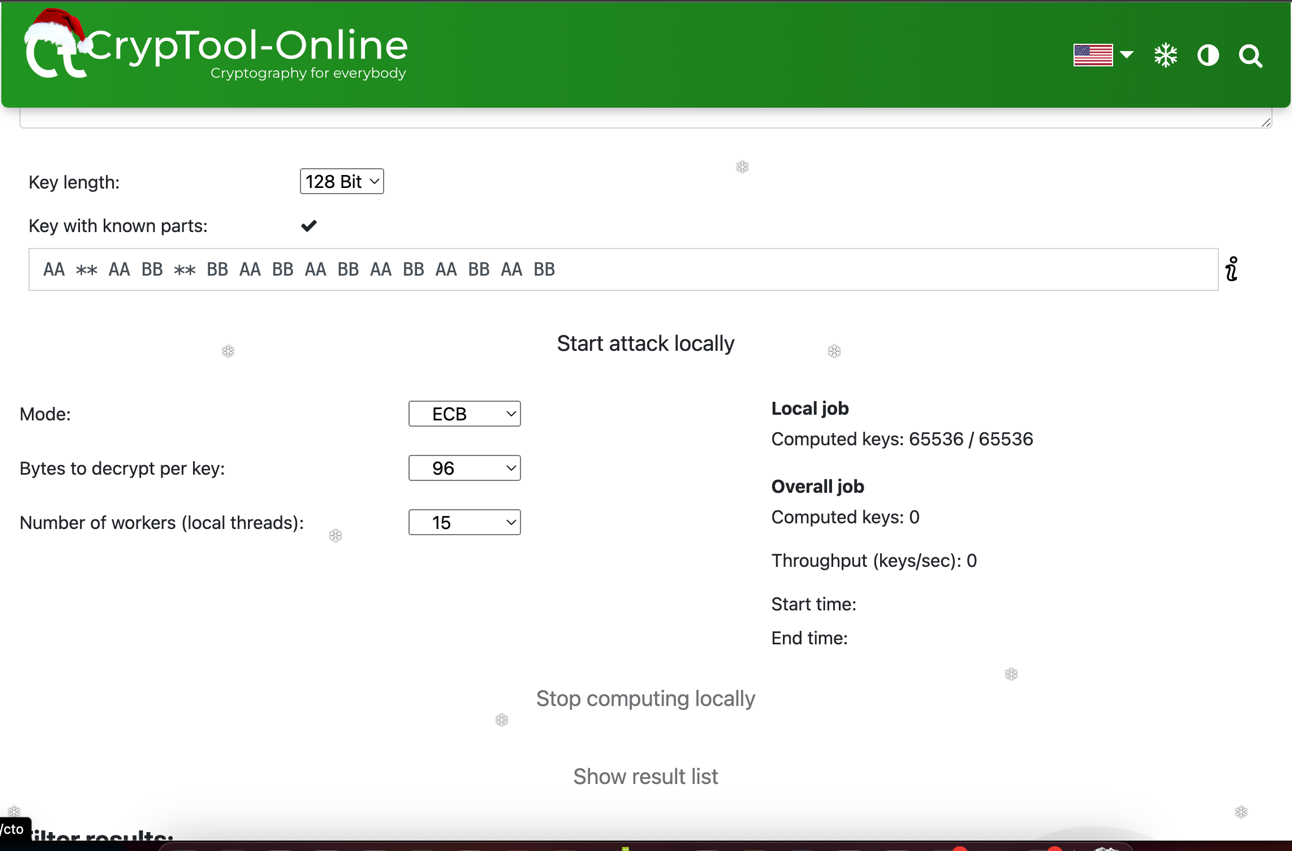
4 stars represent a search space of 16 bit = 216 = 65536 (5 stars = 20 bit = 1,048,576).

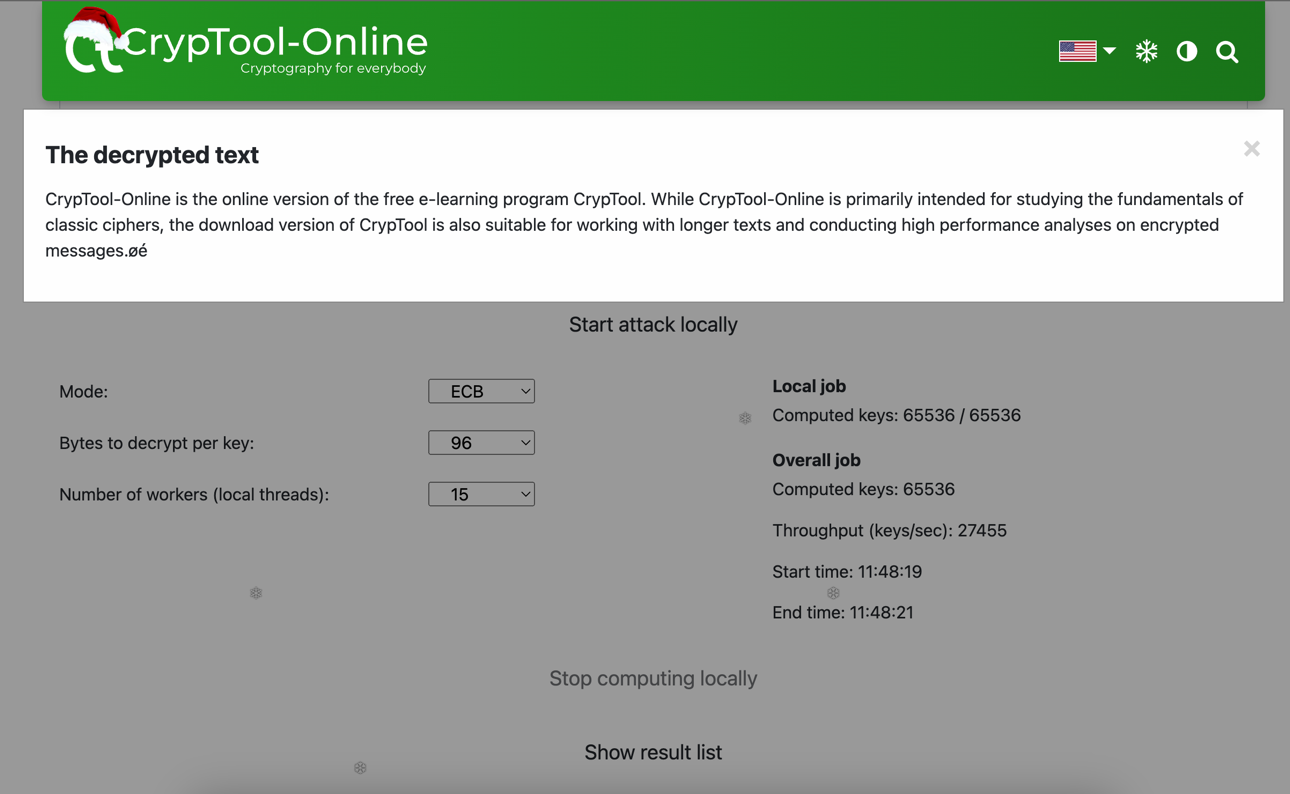
Sample: During the testphase beginning 2017, 15 worker threads exhausted a searchspace of 16 bit on a modern laptop in 1:48 min which means a throughput of 605 keys / sec. On a workstation with 31 workers the same search space was exhausted in 0:34 min (throughput 1915 keys / sec).

You can get further information in an overlay windows when clicking on the symbol i (behind key input field).

**Demonstration:**

****



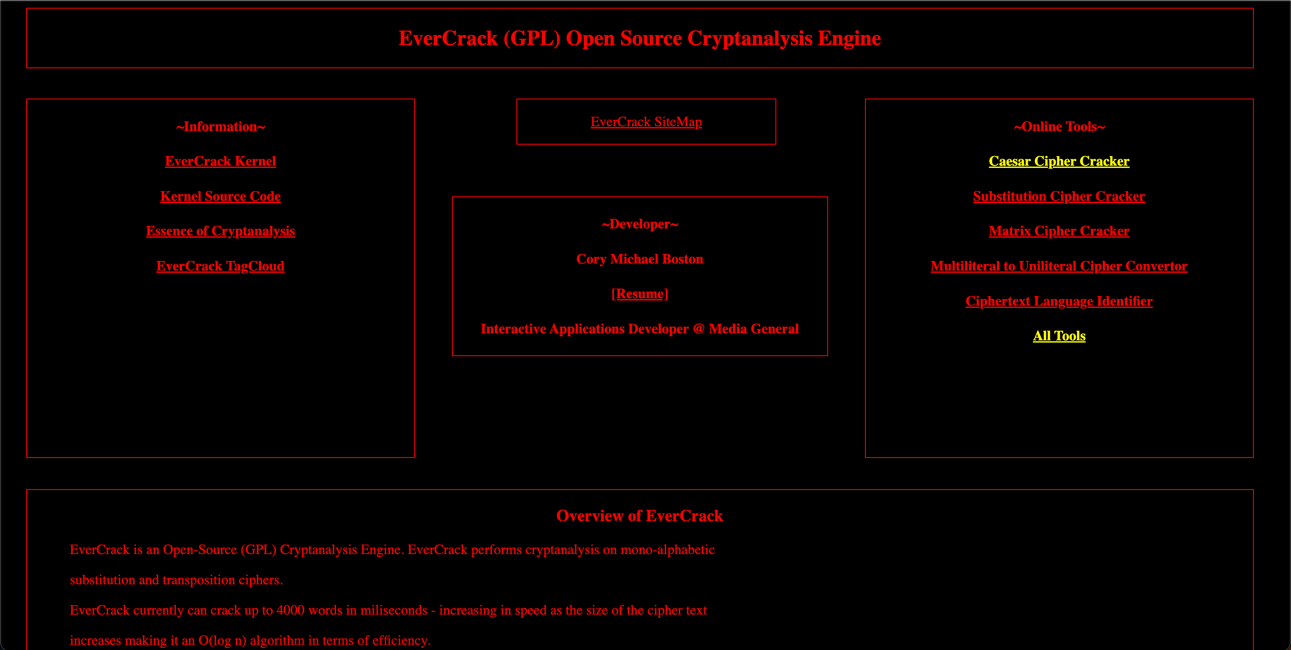


**EverCrack**

An open source GPL software, EverCrack deals chiefly with mono – alphabetic substitution and transposition ciphers. It is a cryptanalysis engine with a multi – language support for English, German, French, Spanish, Italian, Swedish, Dutch and Portuguese. It was initially developed in C language.It is currently concentrating on online web – based applications. Now, the programming is kernel based i.e. deciphering complex ciphers for the kernel.

The overall design goal is to break down complex ciphers systematically into their simplex components for cryptanalysis (by the kernel). The kernel consists of an algebraic design (comparison and reduction) for breaking unilateral, mono – alphabetic ciphers instantaneously. The computational speed is found to be proportional to O (log n).

An EverCrackGUI looks as shown below.

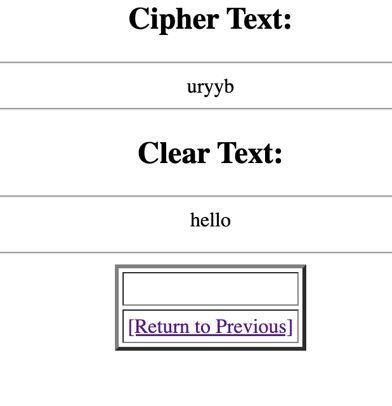


Demo: Let’s attempt a brute force attack to crack Caesar Cipher.

How it works: tests each rot result against words in list of that length.



Output:



**dCode.fr**

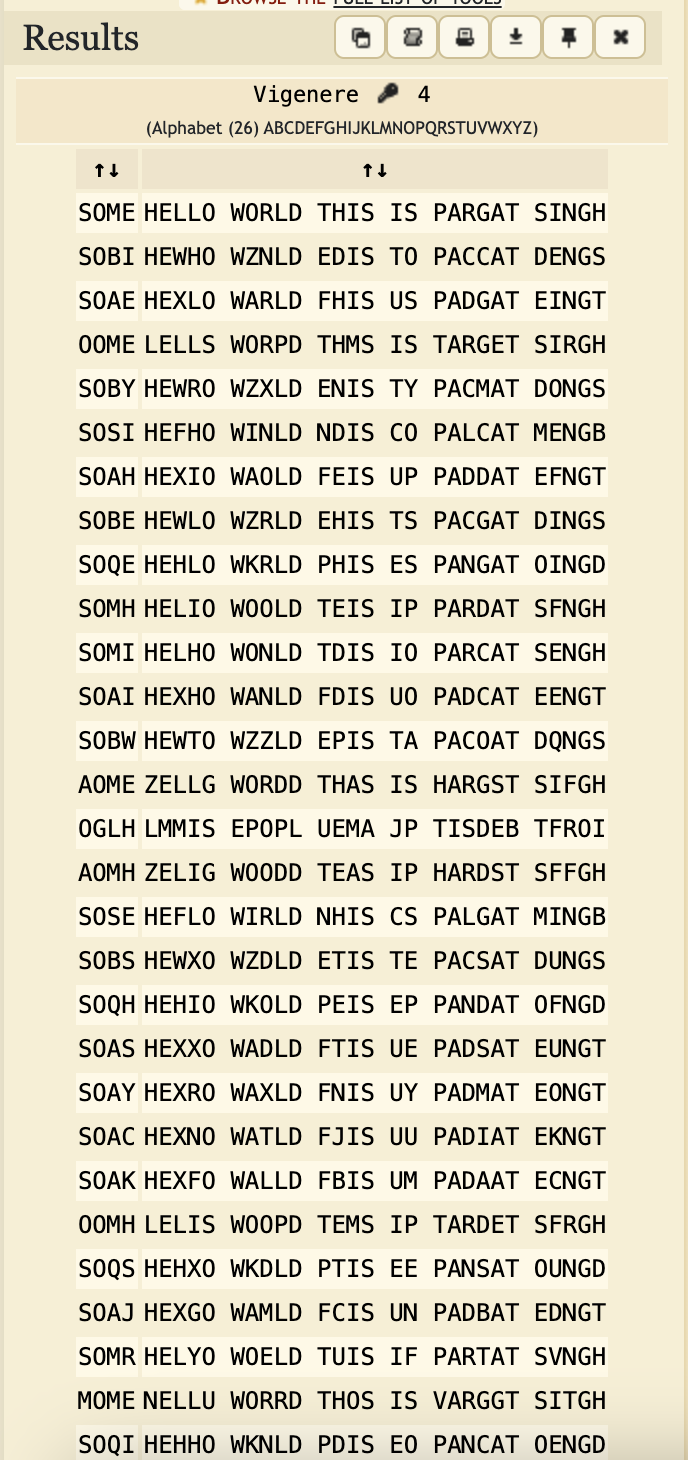
dCode.fr is a set of more than 900 tools to help solve games, puzzles, coded messages, mathematics, etc. dCode is the master of decryption/decoding. Its AI-powered code detector recognizes more than 250 types of ciphers, including the Caesar code , the Vigenère cipher , the Polybius square , as well as dozens of other cryptographic systems . Decrypting messages becomes child's play. dCode offers a huge library of scripts to decode or encode messages with classic cryptography techniques.



Demo : Vigenère cipher

Cipher Text : ZSXPG KAVDR FLAG UW HODKSH EMFUT

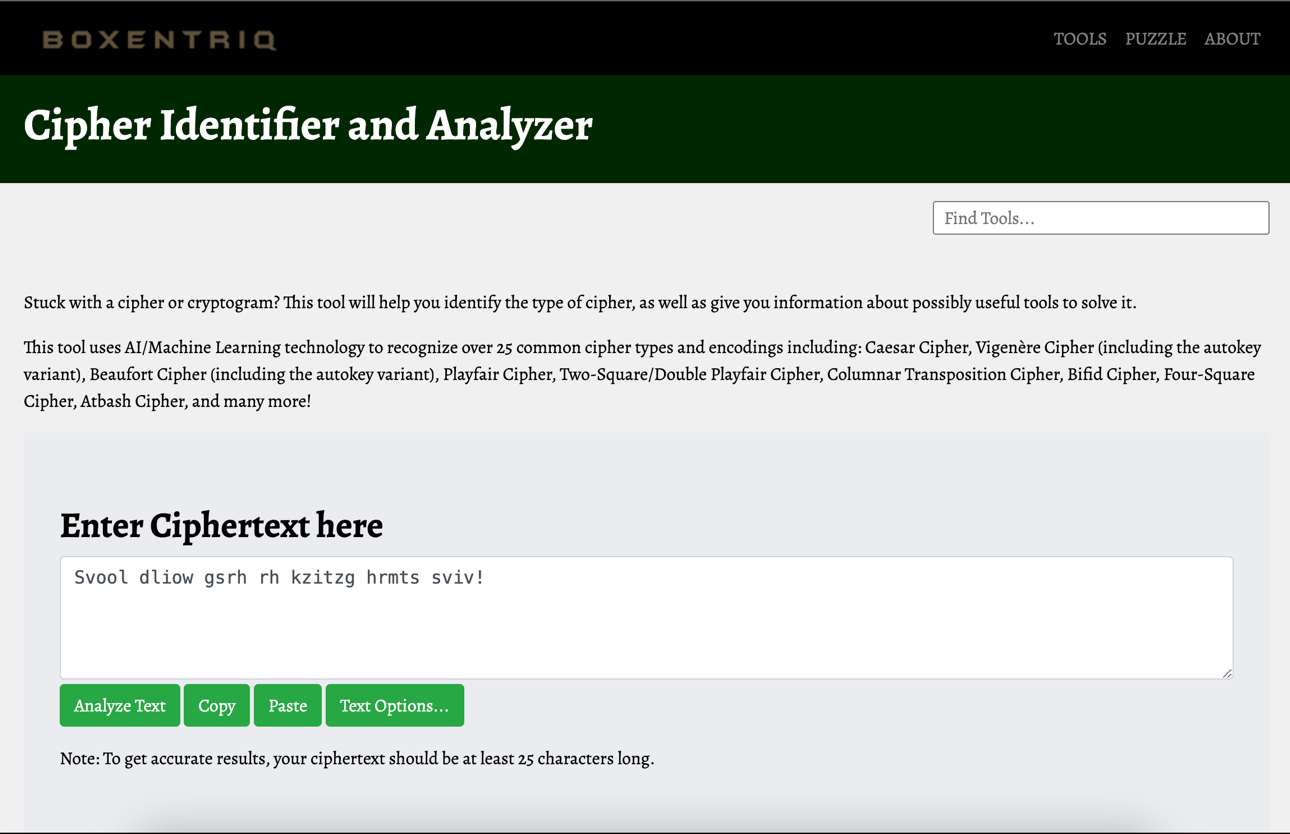




**Boxentriq**

Stuck with a cipher or cryptogram? This tool will help you identify the type of cipher, as well as give you information about possibly useful tools to solve it.

This tool uses AI/Machine Learning technology to recognize over 25 common cipher types and encodings including: Caesar Cipher, Vigenère Cipher (including the autokey variant), Beaufort Cipher (including the autokey variant), Playfair Cipher, Two-Square/Double Playfair Cipher, Columnar Transposition Cipher, Bifid Cipher, Four-Square Cipher, Atbash Cipher, and many more!



**Conclusion:- Learnt to use and implement various cryptographic tools.**

**Postlab questions:  
1.1 Write the points of difference between mono-alphabetic cipher and polyalphabetic cipher.**

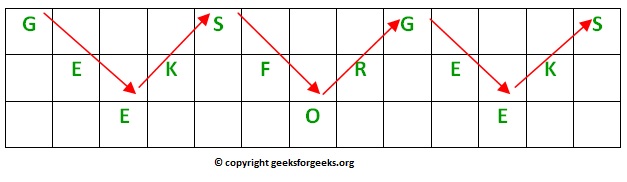
| **SR.NO** | **Monoalphabetic Cipher** | **Polyalphabetic Cipher** |
| --- | --- | --- |
| **1** | **Monoalphabetic cipher is one where each symbol in plain text is mapped to a fixed symbol in cipher text.** | **Polyalphabetic cipher is any cipher based on substitution, using multiple substitution alphabets.** |
| **2** | **The relationship between a character in the plain text and the characters in the cipher text is one-to-one.** | **The relationship between a character in the plain text and the characters in the cipher text is one-to-many.** |
| **3** | **Each alphabetic character of plain text is mapped onto a unique alphabetic character of a cipher text.** | **Each alphabetic character of plain text can be mapped onto ‘m’ alphabetic characters of a cipher text.** |
| **4** | **A stream cipher is a monoalphabetic cipher if the value of key does not depend on the position of the plain text character in the plain text stream.** | **A stream cipher is a polyalphabetic cipher if the value of key does depend on the position of the plain text character in the plain text stream.** |
| **5** | **It includes additive, multiplicative, affine and monoalphabetic substitution cipher.** | **It includes autokey, Playfair, Vigenere, Hill, one-time pad, rotor, and Enigma cipher.** |

**1.2 Explain the working of a rail-fence cipher with the help of an example.**Given a plain-text message and a numeric key, cipher/de-cipher the given text using Rail Fence algorithm.   
The rail fence cipher (also called a zigzag cipher) is a form of transposition cipher. It derives its name from the way in which it is encoded.   
Examples: Encryption

In a transposition cipher, the order of the alphabets is re-arranged to obtain the cipher-text.

* In the rail fence cipher, the plain-text is written downwards and diagonally on successive rails of an imaginary fence.
* When we reach the bottom rail, we traverse upwards moving diagonally, after reaching the top rail, the direction is changed again. Thus the alphabets of the message are written in a zig-zag manner.
* After each alphabet has been written, the individual rows are combined to obtain the cipher-text.

For example, if the message is “GeeksforGeeks” and the number of rails = 3 then cipher is prepared as: 



*.’.Its encryption will be done row wise i.e. GSGSEKFREKEOE*

Decryption

As we’ve seen earlier, the number of columns in rail fence cipher remains equal to the length of plain-text message. And the key corresponds to the number of rails.

* Hence, rail matrix can be constructed accordingly. Once we’ve got the matrix we can figure-out the spots where texts should be placed (using the same way of moving diagonally up and down alternatively ).
* Then, we fill the cipher-text row wise. After filling it, we traverse the matrix in zig-zag manner to obtain the original text.

Implementation:   
Let cipher-text = “GsGsekfrek eoe” , and Key = 3 

* Number of columns in matrix = len(cipher-text) = 13
* Number of rows = key = 3

**1.3 Discuss any three applications of cryptography.**

* Cryptography is widely used to secure communication over the internet. Protocols like HTTPS use cryptographic techniques to encrypt data exchanged between web browsers and servers, ensuring privacy and integrity.

**2. Digital Signatures:**

* Cryptography is employed in creating digital signatures, which authenticate the origin and integrity of digital messages or documents. Digital signatures are crucial in electronic transactions and document verification.

**3. Data Encryption:**

* Cryptography is extensively used to encrypt sensitive data stored on various devices or transmitted over networks. This protects information from unauthorized access and ensures confidentiality. Technologies like BitLocker and FileVault employ encryption to secure data on storage devices.