**Unit 2: Every Bit of the Internet**

**\*\*Instructions:** Please change the text color of your responses to red text. Please organize the endings to each page.

**Project 2.1.2 – Encryption: Keep it Confidential**

1. What are some examples of information you might exchange over the web that you would want to be encrypted?

| Credit card credentials, your social security number, your passwords, and other private info. |
| --- |

1. What is an example of a substitution cipher?

| A caesar cipher and the one-time pad are two examples of a substitution cipher. |
| --- |

1. The following message has been encrypted using Caesar’s cipher:

*Vkliw wkuhh*

| plaintext | 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 |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ciphertext | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | U | V | W | X | Y | Z | A | B | C |

Use the key to decrypt the message.

| Shift three |
| --- |

1. Download and extract the program from mypltw.org. You will not be modifying *main.py* or the other modules used later in this activity. Run the *cipher*program and encrypt a message.
   1. Type **encrypt**at the prompt.
   2. Select a one-sentence message you would like to send. Include your name in the message. Select a key value of 1 to 26. This value will “shift” the letters by the value.
   3. Copy and paste your encrypted message into the shared document provided by your teacher. Remember to provide the “key” with your message.
   4. Using the [shared document](https://docs.google.com/document/d/1C9YbxvaA_ukgDHZPhv05ErxKmePgbvzb6xcYlWjPy58/edit?usp=sharing) link provided by your teacher, select some of the messages your classmates have posted. Use their posted keys to decrypt the messages.
   5. Copy and paste one of the encrypted messages and the decrypted messages in this document.

| pnrfne vf rnfl gb oehgr sbepr oevnaan n  caesar is easy to brute force brianna a |
| --- |

1. Use brute force to decipher the ciphertext below. The ciphertext was created using a Caesar-like cipher that shifts the letters of the alphabet.

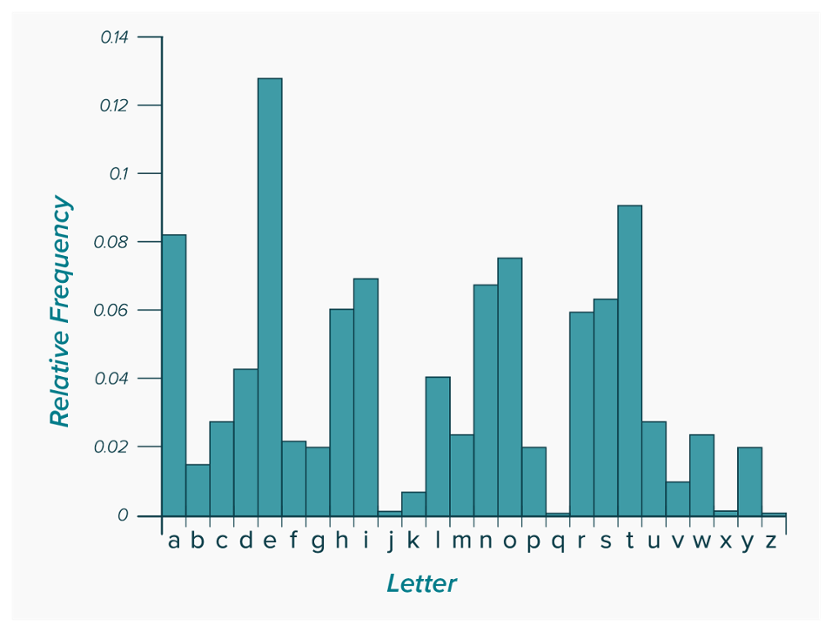
*Guvf vf n rapelcgrq zrffntr. Lbh jvyy arire or noyr gb ernq vg!*

Use your *cipher*program to decrypt the ciphertext using each of the possible twenty-six keys. Run your program again, but type **brute**when prompted to choose encrypt or decrypt.  Paste the ciphertext into the prompt.

| This is an encrypted message. You will never be able to read it! |
| --- |

Can you easily identify the message based on the output? Why or why not?

| Yes, as there are only 26 possible messages, making it easy to look through all of them and find the original message. |
| --- |

1.   
   Study the histogram above.  
   Which letters are the most commonly used letters in English language plaintext?

| e, t, a, o, i, n, s, h, and r are the most commonly used letters in the English language plaintext. |
| --- |

1. You will use frequency analysis and some common English language conventions to solve a popular puzzle known as a cryptogram. A cryptogram is a type of puzzle that consists of a short piece of encrypted text. Cryptograms use substitution ciphers where each letter is replaced by a different letter. To solve a cryptogram, you must recover the original lettering.

*vypb yoi vhss xqou nc urq rhss uj dqupr y cyhs jd xyuql. vypb dqss ijxo yoi gljbq rhm pljxo, yoi vhss pyeq unegshok yduql. nc vypb kju, yoi rjeq ihi ulju, ym dymu ym rq pjnsi pycql, rq xqou uj gqi uj eqoi rhm rqyi, xhur thoqkyl yoi gljxo cycql.*

Create a document or spreadsheet that lists all the letters of the alphabet. You will use this to keep track of the substitutions you make throughout this exercise. You may need to reference this to undo some of your changes if they happen to be incorrect.

| plaintext | 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 |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ciphertext |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

Run the frequency program that you downloaded at the beginning of this activity. When prompted to enter the string to analyze, paste the encrypted cryptogram into the program.

Which letter occurs the most frequently?

| y |
| --- |

Second most frequently?

| q |
| --- |

How could you use this information to decrypt a message?

| The letter frequencies can be used to guess letters, as the most common letter is usually e and the second most common letter is usually t, a, o, i, or n. |
| --- |

1. Decipher the cryptogram, continuing to use the letter\_swapper program and the letter frequency analysis information to systematically deduce the original plaintext.
   1. Record all the swaps you make on a piece of paper, a text document, or a spreadsheet.

y -> a

u -> t

r -> h

q -> e

d -> f

m -> s

s -> l

h -> i

p -> c

x -> w

l -> r

n -> u

j -> o

i -> d

o -> n

t -> v

k -> g

c -> p

g -> b

v -> j

b -> k

e -> m

* 1. If you want to undo a previous replacement, simply enter the original letter and replace it with itself. For example, if you substituted "y" for "e," you can select "y" and "y" to revert it back to its original letter.

Jack and Jill went up the hill to fetch a pail of water. Jack fell down and broke his crown, and Jill came tumbling after. Up jack got, and home did trot, as fast as he could caper, he went to bed to mend his head, with vinegar and brown paper.

1. Run the generate\_keys program you downloaded at the beginning of this activity. Record all three keys that are provided. The first two numbers are randomly selected prime numbers used to generate the keys. The public key and modulus are to be shared in the document discussed by your teacher. (The modulus is used in RSA algorithm to link the two keys together.) Keep the private key for yourself.
   1. Publish your public key and modulus to the shared document provided by your teacher.  
      [Google Sheets Link](https://docs.google.com/spreadsheets/d/1czpDtKreqVMNpPtDcsuGnOgLRecaQkWmFHC_jqptP9Y/edit?usp=sharing)
   2. Select another student you would like to send a message to. Copy their public modulus and modulus.
   3. Run rsa\_encrypt and use the public key that was shared with you to encrypt a message to send to your partner.
   4. Decrypt messages sent to you using the rsa\_decrypt program and your private key. You will have to paste the message without brackets.

| 30519, 20772, 4271, 9285, 13656, 32764, 25378, 4271, 32764, 27743, 32764  “hi kyle lol” |
| --- |

Public key: 18305

Private key: 29537

Modulus: 35011

1. Encrypt a message with your private key. Include your name in the message and a short phrase. Publish the encrypted message along with your public key to the shared document provided by your teacher. Indicate that it is your public key.  
   1. Decrypt messages sent by other classmates using the published public keys.
   2. Can you verify whom these messages came from? Why or why not?

| You can verify who the messages came from because messages that can be decrypted with the public key have to have been written with the private key, meaning that the author must have had access to the private key. |
| --- |

1. Certificate Authorities we will do as a class.