Encryption (non-PKE)

Presented By:

Christine Marra
Adam Prado
Wayne Tobias
Yenmin Young
Vanessa Zou (Qianhui)

Break The Code:

Ygneqog vq qwt Gpetarvkqp Rtgugpvcvkqp!

Virtual Wheel: Cipher Wheel Use a shift of 2.

What is Encryption?

Encryption is the process of transforming information into a form that is unreadable by anyone other than those the information is intended for.

Private Key Encryption:

- symmetric encryption
- a single private key can encrypt and decrypt information

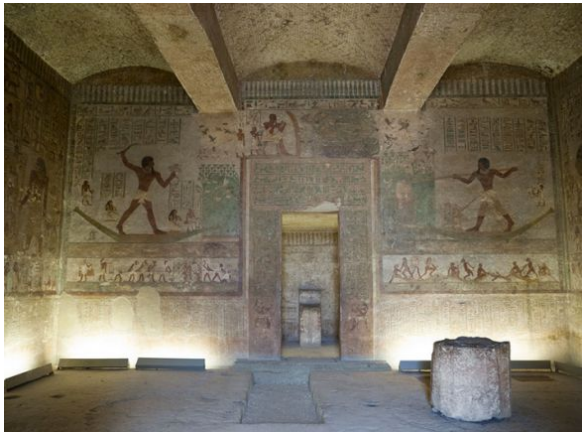
Public Key Encryption:

- PKE
- asymmetric encryption
- uses two keys – one private and one public. The public key is distributed, whereas the private key is never shared.

Source: https://koolspan.com/private-key-encryption/

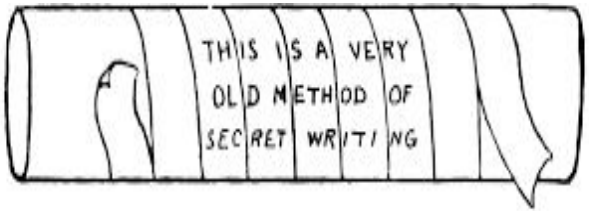
History of Encryption

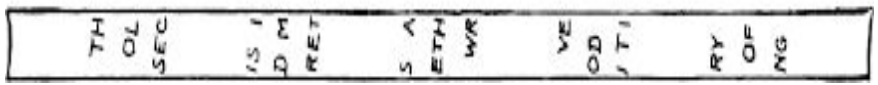
- Earliest written evidence of encryption can be traced to ancient Egypt.
- Tomb of nobleman Khnumhotep II, 1,900 B.C. contained a script recording his deeds in life and some unusual hieroglyphs that obscured the original meaning of the text.



History of Encryption

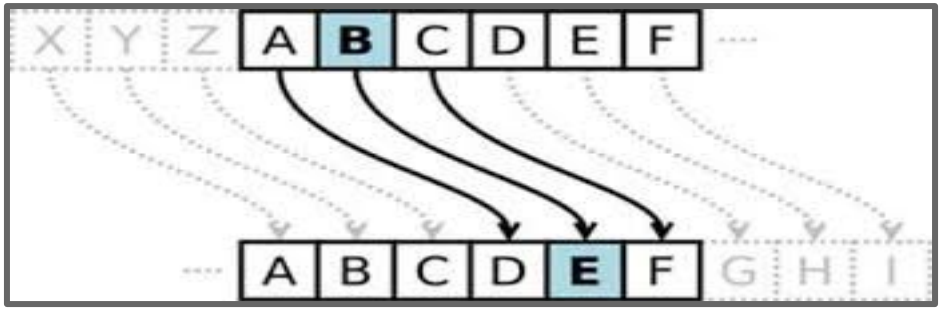
- The Spartans, 5 BC, developed a device called a Scytale.
- It is a type of Transposition Cipher.
- A messenger would carry a strip of parchment, which was meaningless until it was wrapped around a Scytale of the same diameter. Source:tps://tresorit.com/blog/the-history-of-encryption-the-roots-of-modern-day-cyber-security/





History of Encryption

- The Romans, 100 BC, developed a Shift Cipher (Caesar Cipher).
- It is a type of substitution cipher.
- Uses the normal sequence of the alphabet but *shifts* letters a fixed number of letters further down the alphabet.
- Decrypted in 800 AD by Al-Kindi, father of cryptanalysis, by looking at the frequency of letters in the encrypted message to determine the shifting rule.



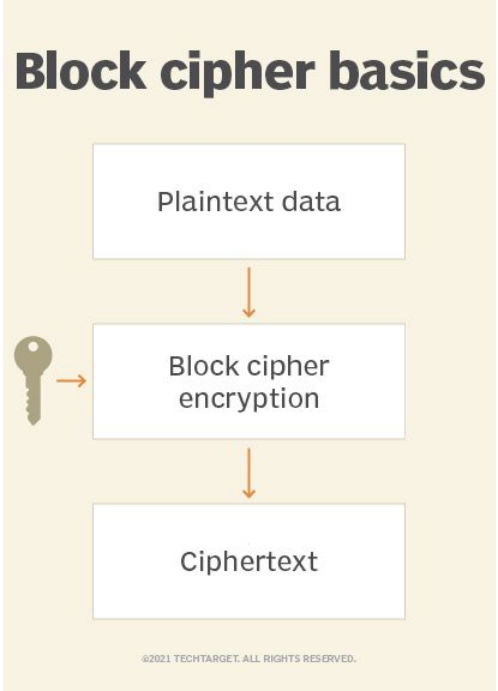
History of Encryption

- Modern cryptography emerged during World War II with the emergence of machine and electromechanical encryption/decryption.
- Arthur Scherbius invented a changing substitution cypher, or a polyalphabetic substitution cypher, known as the German Enigma Machine.
- It was made using Rotors and Plugboards.
- One or more of the rotors moved after each key press, depending on the settings.

Block Ciphers

A method of encrypting data in blocks to produce ciphertext using a cryptographic key and algorithm. Uses symmetric key and algorithm to encrypt and decrypt.

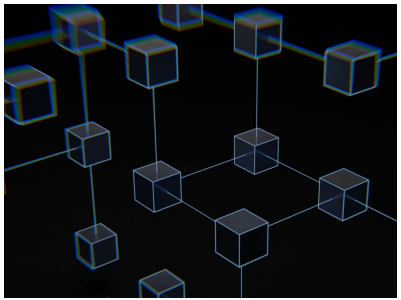
- One of the two main categories of symmetric encryption
- Plaintext is broken into fixed-sized blocks and algorithm operates on each block independently to convert into ciphertext using a key
    - Usual block sizes are 64 bits, 128 bits, and 256 bits.
- An ideal block cipher uses the permutation process to create a key where brute force is difficult to use to break the key
- An initialization vector is added to the plaintext to avoid repetition of binary sequence
- Found in computer communications, anywhere in cyber security



Block Ciphers: Modes of Operation

Blocks of plaintext with more or less blocks are encrypted separately since messages are encrypted when it is the same size as the block length.

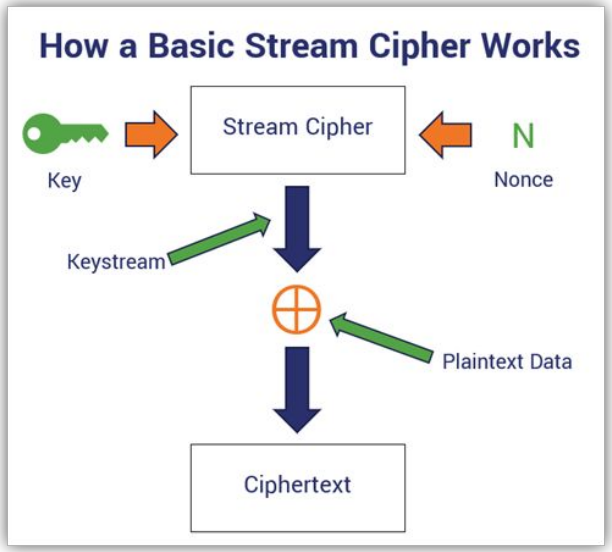
- **Electronic codebook (ECB)**
- **Cipher block chaining (CBC)**
- **Ciphertext feedback (CFB)**
- **Output feedback (OFB)**
- **Counter (CTR)**



Popular block ciphers are Data Encryption Standard (DES), Triple DES, Advanced Encryption Standard (AES), Blowfish, and Twofish.

Stream Ciphers

Breaks a plaintext message down into single bits instead of in blocks, which then are converted individually into ciphertext using key bits.



Quick info on Block and Stream Ciphers

Block Ciphers	Stream Ciphers
Symmetric key ciphers that encrypt and decrypt data in fixed-size blocks.	Symmetric key ciphers are stateful ciphers that encrypt and decrypt data bit-by-bit.
Slower processing.	Faster processing.
Require more resources.	Require fewer resources.
Can take on stream cipher properties through certain modes of operation.	Cannot take on block cipher properties.
Rely on stateless and stateful modes of operation, which include ECB, CBC, CFB, OFB, CTR, GCM, and XTS.	Can be synchronous or asynchronous.
Used nearly everywhere.	Used for some data in-transit encryption, including in some SSL/TLS cipher suites.

Multiple Level Encryption

Process of encrypting an already encrypted message one or more times, using the same or different algorithm.

- Use keys that are statistically independent for each layer
- The Rule of Two is data security principle that specifies two completely independent layers of cryptography to protect data
- It is practiced in the NSA's secure mobile phone, Fishbowl. It uses two layers of encryption protocols, IPsec and Secure Real-time Transport Protocol, to protect voice communications.
    - To ensure protection, components from the same manufacturer must provide evidence that the implementations of the two components are independent of one another.
    - Another way is to implement each layer using components produced by different manufacturers.

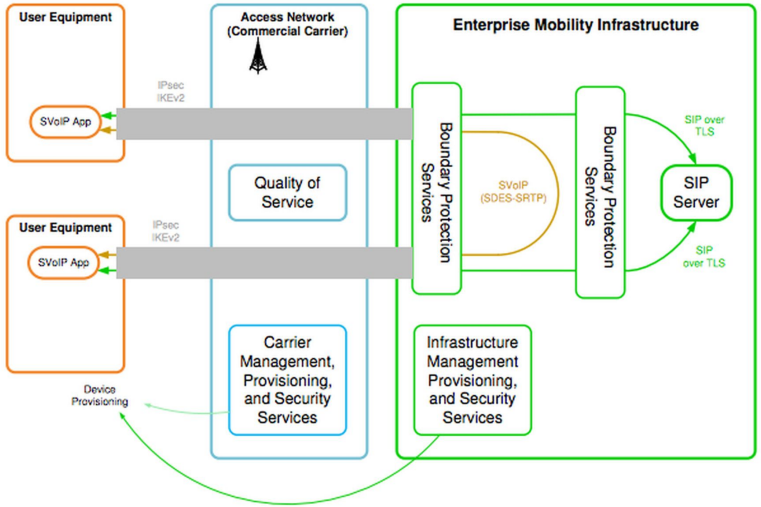


Figure 2-2 Basic Segments of Secure VOIP

Encryption and Decryption

Encryption is the process of converting plaintext (readable data) into unreadable ciphertext (encrypted data) using a secret key. This ensures that the data can only be read by someone who has the key to decrypt it.

Encryption and Decryption

Decryption is the reverse process of encryption. It's the conversion of ciphertext back into its original form (plaintext), using the same secret key that was used to encrypt it.

Encryption and Decryption

As you saw in the pre-work, the Caesar Cipher is a simple encryption technique named after Julius Caesar, who used it to communicate with his officials. In this technique, each letter in the original message (plaintext) is shifted a certain number of places down the alphabet to form the encrypted message (ciphertext).

Encryption and Decryption

This technique is not very secure by modern standards, as it's easily broken by a process called frequency analysis. However, it's still a good example to illustrate the basic idea of substitution ciphers, which are encryption techniques that replace each letter in the plaintext with another letter, symbol, or number.

We'll explore frequency analysis during the second part of this lesson.

Pre-Code

A few key points

1) Define algorithm being used
2) Convert to .upper()
3) Ignore non str objects
4) Modulo % 26
5) Decrypt code is opposite direction of Encrypt code
6) Refactoring - repetitive code, dedicated functions

The Algorithm

In this program we will write an algorithm that uses the arguments **key** and **message**. They will hold the two pieces of information we need to focus on. What is the shift in the key and what is the original message.

Another concept we will use in this lab is to convert all of our string text to uppercase by using the **.upper()** string method. The uppercase alphabet will be stored in a variable and we can loop through this list.

Python ideas

Convert to .upper()
The .upper() method is a built-in function in Python that can be used with a string to convert all its characters to uppercase letters. In other words, it takes a string and makes all the letters in that string capital letters.

if we have the string "hello world" and you apply the .upper() method to it, the result would be "HELLO WORLD". This method can be useful when you need to compare strings in a case-insensitive manner or when you want to display text in a consistent way regardless of the user's input.

Python ideas

Ignore non str objects

To simplify our task for today, we will ignore any non string characters in our code. So periods, commas, numbers, etc will not be encoded.

Think about other methods for dealing with special characters.

Do we need all non string characters or is there an alternative?

The Shift

Handling loop-arounds. This is going to solve the problem we encounter at the end of the alphabet. If our index lands on a number greater than 26, we want to subtract 26. For example the index 28 would become the index number 2.

We will use modulo to help us. Any number greater than 26 will simply be reduced to a number from 0 - 25. Exactly what we need.

26%26 = (1 * 26 + 0) %26 = 0
27%26 = (1 * 26 + 1) %26 = 1
28%26 = (1 * 26 + 2) %26 = 2

Cleaning up your code

Consider these two concepts for your final code.

1) Decrypt code (+) is opposite direction of Encrypt code (-)
   You will write the code to perform the opposite function.

2) Refactoring - repetitive code, dedicated functions
   Clean up your code to remove repetitive code or decide where a function can be reused.

Python

Alternative options:

1) Code together
2) Get code at the end

Shift over to Python code

Frequency Analysis

Frequency Analysis is the study of frequency of letters or groups of letters in a ciphertext. This can be used to help deduce the keyshift for a Caesar Cipher.

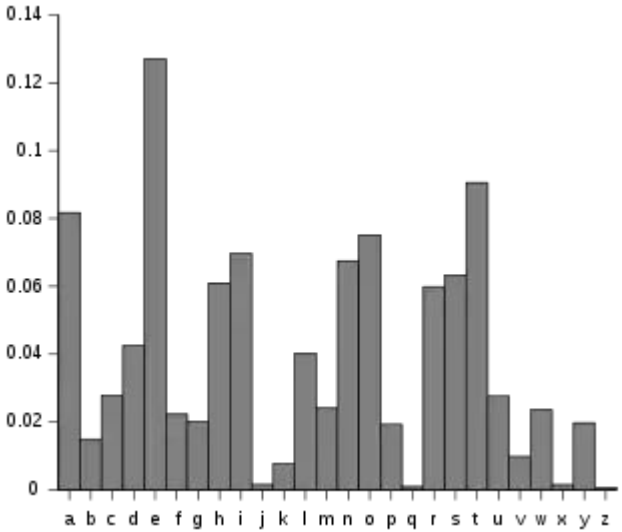
Check out the Count Letter Frequency tool

Most common letters:

E, T, A, O, I, N, S, R, H

Least common letters:

J, Q, X, Z

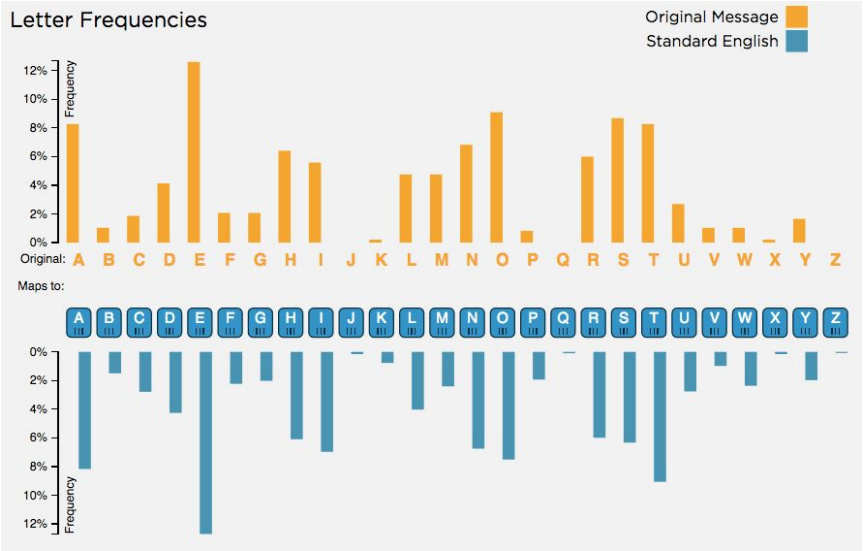


Letter Frequency Analysis...in Action!

Give Yenmin an encoded text from Cesar Cipher and she'll guess which key was used. (minimum 5000 characters or 800 words)

[Letter Frequency Grapher]

[Code.org Letter Analysis Widget]



Homework!

1) Decode a message that was encoded using a simple shift/caesar cipher.

- There are two starter code options (with and without punctuation)

2) Encode a message of your own using any method of encryption besides a simple shift/caesar cipher.

https://docs.google.com/document/d/1FTBp-1Gl4wKYR1aVYUlkZKwJlCvg4Ugt7Sc8juqDfJc/edit?usp=sharing

Homework!

A) Attempt to write code for 1 of the two parts of the assignment (encode or decode).

- Link to a repl.it file

- Add comments to explain

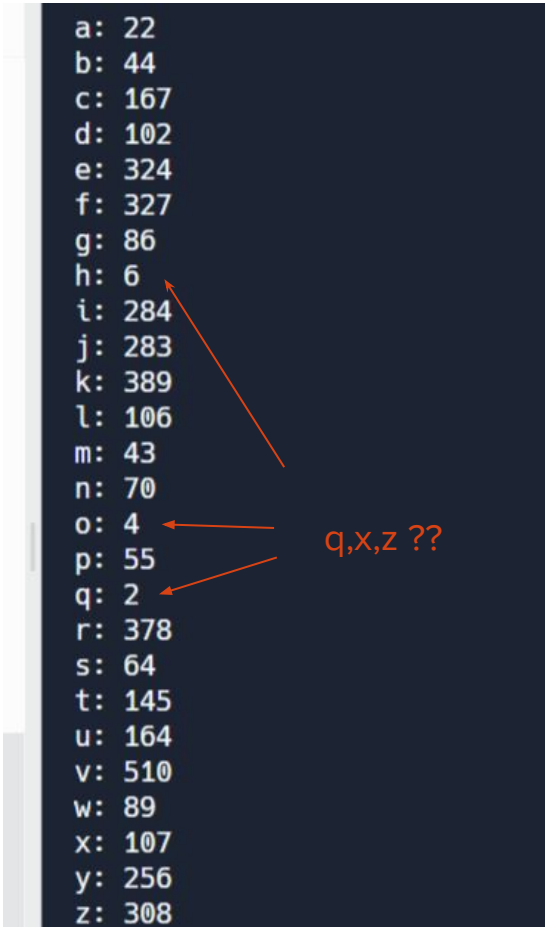
B) Explain a detailed algorithm in words for the other part.

- .txt, .pdf, .md

Homework!

```
message = MABL VTG'M IHLLBUER PHKD!
NBCM WUH'N JIMMCVFS QILE!,  shift = 1
OCDN XVI'O KJNNDWGT RJMF!,  shift = 2
PDEO YWJ'P LKOOEXHU SKNG!,  shift = 3
QEFP ZXK'Q MLPPFYIV TLOH!,  shift = 4
RFGQ AYL'R NMQQGZJW UMPI!,  shift = 5
SGHR BZM'S ONRRHAKX VNQJ!,  shift = 6
THIS CAN'T POSSIBLY WORK!,  shift = 7
UIJT DBO'U QPTTJCMZ XPSL!,  shift = 8
VJKU ECP'V RQUUKDNA YQTM!,  shift = 9
WKLV FDQ'W SRVVLEOB ZRUN!,  shift = 10
XLMW GER'X TSWWMFPC ASVO!,  shift = 11
YMNX HFS'Y UTXXNGQD BTWP!,  shift = 12
ZNOY IGT'Z VUYYOHRE CUXQ!,  shift = 13
AOPZ JHU'A WVZZPISF DVYR!,  shift = 14
BPQA KIV'B XWAAQJTG EWZS!,  shift = 15
CQRB LJW'C YXBBRKUH FXAT!,  shift = 16
DRSC MKX'D ZYCCSLVI GYBU!,  shift = 17
ESTD NLY'E AZDDTMWJ HZCV!,  shift = 18
FTUE OMZ'F BAEEUNXK IADW!,  shift = 19
GUVF PNA'G CBFFVOYL JBEX!,  shift = 20
HVWG QOB'H DCGGWPZM KCFY!,  shift = 21
IWXH RPC'I EDHHXQAN LDGZ!,  shift = 22
JXYI SQD'J FEIIYRBO MEHA!,  shift = 23
KYZJ TRE'K GFJJZSCP NFIB!,  shift = 24
```

```
a: 22
b: 44
c: 167
d: 102
e: 324
f: 327
g: 86
h: 6
i: 284
j: 283
k: 389
l: 106
m: 43
n: 70
o: 4
p: 55
q: 2
r: 378
s: 64
t: 145
u: 164
v: 510
w: 89
x: 107
y: 256
z: 308
```



Q & A