



Universidad Politécnica
de Madrid

**Escuela Técnica Superior de
Ingenieros Informáticos**



Grado en Matemáticas e Informática

Trabajo Fin de Grado

Sistemas de Criptografía a lo Largo de la Historia

Autor: David González Labrada
Tutora: Águeda Mata Hernández

Madrid, Junio 2024

Este Trabajo Fin de Grado se ha depositado en la ETSI Informáticos de la Universidad Politécnica de Madrid para su defensa.

Trabajo Fin de Grado
Grado en Matemáticas e Informática

Título: Sistemas de Criptografía a lo Largo de la Historia
Junio 2024

Autor: David González Labrada

Tutor: Águeda Mata Hernández

Departamento de Matemática Aplicada a las Tecnologías de la
Información y las Comunicaciones
Escuela Técnica Superior de Ingenieros Informáticos
Universidad Politécnica de Madrid

Resumen

Este Trabajo de Fin de Grado muestra un estudio de los métodos criptográficos desde la Antigüedad hasta la llegada de los primeros ordenadores. Se ha llevado a cabo una revisión bibliográfica y un estudio histórico de numerosos textos con el objetivo de comprender la evolución de la disciplina y su importancia en el transcurso de la historia. En este contexto, se analizan grandes hitos de la disciplina como la máquina Enigma y el criptoanálisis realizado por Alan Turing y Marian Rejewski durante la Segunda Guerra Mundial, así como el impacto del telegrama Zimmermann en la Primera Guerra Mundial. De forma adicional, se han implementado diversos métodos criptográficos en Python para permitir al usuario el cifrado y descifrado de mensajes a través de una aplicación, proporcionando una comprensión práctica de los conceptos teóricos presentados.

Abstract

This Final Degree Project presents a comprehensive study of cryptographic methods from ancient times to the advent of the first computers. A bibliographic review and historical analysis of numerous texts have been conducted with the objective of understanding the evolution of the discipline and its significance throughout history. Within this context, key milestones in the field are examined, including the Enigma machine and the cryptanalysis efforts by Alan Turing and Marian Rejewski during World War II, as well as the impact of the Zimmermann Telegram in World War I. Furthermore, various cryptographic methods have been implemented in Python, enabling users to encrypt and decrypt messages through an application, providing a practical understanding of the theoretical concepts discussed.

Tabla de contenidos

1. Introducción	1
2. Nociones básicas de criptografía	3
2.1. Esteganografía	3
2.2. Criptografía	5
3. Criptografía desde la Antigüedad hasta la Primera Guerra Mundial	9
3.1. Primeros ejemplos	9
3.2. El método criptográfico de la escítala lacedemonia	10
3.3. Cifrado Atbash	11
3.4. Eneas	12
3.5. Polibio	13
3.6. Julio César	14
3.7. Criptoanálisis árabe	15
3.8. Edad Media	18
3.9. Leon Battista Alberti	19
3.10 Nomenclátor	21
3.11 Johannes Trithemius	22
3.12 Pavía Girolamo Cardano	24
3.13 Giovan Battista Bellaso	25
3.14 Giovanni Batista Della Porta	25
3.15 Blaise de Vigenère	26
3.16 Cifrado Pigpen	27
3.17 La Gran Cifra de París	28
3.18 Julio Verne	29
3.19 Lyon Playfair y Charles Wheatstone	32
3.20 Babbage y Kasinski	36
3.21 Samuel Morse	37
3.22 Auguste Kerckhoffs	39
3.23 La sala 40 y el telegrama Zimmermann	39
3.24 ADFGX	41
4. Criptografía en la Segunda Guerra Mundial	45
4.1. Gilbert Vernam	45
4.2. Los cifradores de rotación	47
4.3. La máquina Enigma	49

TABLA DE CONTENIDOS

4.4. El criptoanálisis polaco de la máquina Enigma	53
4.4.1. Estudio de las permutaciones	55
4.4.2. Hojas de Zygaliski	66
4.4.3. Método del reloj	69
4.4.4. Método ANX	70
4.4.5. Bombas	70
4.5. Bletchley Park	72
4.6. Alan Turing	73
4.7. Las máquinas Lorenz	77
4.7.1. Criptoanálisis y descriptado	78
4.7.2. Colossus	81
4.8. Criptografía y computación	82
5. Codificación	85
5.1. Escítala o cifrado de transposición	85
5.2. Cifrado Atbash	87
5.3. Cifrado de Polibio	88
5.4. Cifrado Cesar	90
5.5. Análisis de frecuencia	92
5.6. Cifrado Vigenère	93
5.7. Cifrado Playfair	96
5.8. Código Morse	98
5.9. Cifrado ADFGX y ADFGVX	99
5.9.1. ADFGX	99
5.9.2. ADFGVX	102
5.10Máquina Enigma	103
5.10.1Máquina Enigma M3	103
5.10.2Máquina Enigma M4 o SHARK	106
5.11Interfaz gráfica	107
6. Resultados y conclusiones	113
7. Análisis de impacto	119
Bibliografía	121
Anexos	125

Capítulo 1

Introducción

La criptografía es una disciplina que está presente en la vida de todos nosotros, aunque a menudo pase desapercibida. Palabras como "encriptar" o "cifrar" han entrado en el vocabulario popular, ¿quién no ha visto el mensaje de "cifrado de extremo a extremo" al utilizar aplicaciones de mensajería tan comunes como WhatsApp? Este concepto, que parece tan cotidiano hoy en día, es primordial para garantizar la seguridad y privacidad de nuestras comunicaciones digitales.

A pesar de que hoy en día asociamos la criptografía principalmente con la seguridad informática, su origen se remonta a mucho más atrás. Desde tiempos antiguos, la necesidad de proteger la información ha sido constante, todo ser humano tiene secretos y desea ocultarlos. Desde los cifrados sencillos utilizados en la Roma antigua hasta los complejos algoritmos modernos, la criptografía ha evolucionado en respuesta a las necesidades de seguridad de cada época.

En la antigüedad, se utilizaban métodos rudimentarios para cifrar mensajes, pero incluso estos primeros esfuerzos muestran la importancia que las civilizaciones le dieron a la protección de la información confidencial. La criptografía fue esencial en contextos militares, donde la capacidad de mantener en secreto los planes y comunicaciones podía determinar el éxito o el fracaso de una misión.

Con el paso del tiempo, la criptografía no solo continuó aplicándose en un contexto bélico, sino que también comenzó a jugar un papel crucial en el comercio y la política. A medida que las sociedades se volvieron más interconectadas, la necesidad de proteger la información aumentó. En la Edad Media y el Renacimiento, la criptografía se convirtió en una herramienta indispensable para los comerciantes y diplomáticos que necesitaban asegurar sus comunicaciones frente a posibles interceptaciones.

La llegada de la era moderna trajo consigo avances tecnológicos que transformaron la criptografía de manera radical. La invención de dispositivos mecánicos y, posteriormente electrónicos, permitió la creación de sistemas de cifrado mucho más complejos. Durante el siglo XX, los avances en matemáticas y la aparición de la computación llevaron a un salto cualitativo en la criptografía. La Segunda Guerra Mundial marcó un punto de inflexión, no solo por el uso intensivo de

Capítulo 1. Introducción

la criptografía, sino también por los esfuerzos sin precedentes para romper los cifrados enemigos que cambiarían el rumbo de la guerra.

Los principales objetivos de este trabajo son estudiar los métodos criptográficos desde sus inicios hasta la llegada de los primeros ordenadores y explorar la implementación de algunos de estos métodos. Además de esto, también se pretende estudiar la carrera entre los criptógrafos y los criptoanalistas, examinando cómo cada avance en la criptografía desencadenaba nuevas técnicas y estrategias en el campo del criptoanálisis, creando una dinámica de innovación y desafío continuo. A través de este recorrido histórico y técnico, se pretende comprender cómo la criptografía ha llegado a ser lo que es hoy en día y la importancia que ha tenido en el transcurso de la historia. Este estudio se realiza con el apoyo de "Historia de la criptografía" de Manuel J. Prieto [1], "La criptografía" de Luis Hernández Encinas [2], "Matemáticos, espías y piratas informáticos" de National Geographic [3], el artículo "Criptología Nazi. Los códigos secretos de Hitler" de José Manuel Sánchez Muñoz [4] y "Criptografía sin secretos en Python" de David Arboledas Brihuega [5].

Capítulo 2

Nociones básicas de criptografía

Para tener una comprensión más sólida de los métodos de ocultación empleados en la historia de la criptografía es fundamental familiarizarse con varios conceptos básicos. Estos conceptos proporcionarán un marco de referencia necesario para comprender cómo se aplicaron diferentes métodos de ocultación a lo largo de la historia. En el contexto de este trabajo, se referirá con el término "texto claro" o "mensaje en claro" al mensaje original o la información que se desea comunicar. Es decir, es la información que se desea proteger o transmitir de manera segura.

Para comenzar, se deben distinguir los siguientes métodos para proteger dicho texto claro:

2.1. Esteganografía

El enfoque ingenioso de la esteganografía radica en ocultar la existencia del mensaje, convirtiéndose en una herramienta muy poderosa para todos aquellos que busquen mantener la confidencialidad de su comunicación. El objetivo es que la información que se trata de ocultar y transmitir pase desapercibida ante cualquier intento de detección por parte de un tercero no autorizado.

El término "esteganografía" tiene sus raíces en el griego antiguo, combinando las palabras "steganos" (cubierto, que protege) y "graphos"(escritura). Esta unión de conceptos resalta la naturaleza de esta práctica, donde el mensaje se oculta ingeniosamente dentro de otro contexto aparentemente inocente.

Se pueden distinguir dos tipos de esteganografía:

1. Esteganografía técnica:

Se conoce por esteganografía técnica a toda aquella que haga uso de cualquier método que no sea un texto para ocultar el mensaje real. Uno de los ejemplos más comunes que vivimos en nuestro día a día es la impresión en los billetes de un patrón especial de pequeños puntos denominado constelación de EURión. Es utilizado como medida de falsificación al poder ser detectada por un escáner o impresora, impidiendo su impresión o escaneo.



Figura 2.1: Billeto de 10€.

2. Esteganografía lingüística

Por otro lado, se llama esteganografía lingüística a aquella en la que es un texto lo que oculta el mensaje que se quiere transmitir. Hay diversas formas de hacerlo, pero una de las más famosas son los acrósticos, es decir, hacer que las letras iniciales de cada línea o frase formen el mensaje oculto. El ejemplo más conocido aparece representado en la célebre obra *La Celestina* de Fernando de Rojas:

[...]

Fuertes más que ella por cebo la llevan:

En las nuevas alas estaba su daño.

Razón es que aplique a mi pluma este engaño,

No disimulando con los que arguyen;

Así que a mí mismo mis alas destruyen,

Nublosas e flacas, nacidas de hogaño.

Donde ésta gozar pensaba volando,

O yo aquí escribiendo cobrar más honor,

De lo uno y lo otro nació disfavor:

Ella es comida y a mí están cortando

Reproches, revistas e tachas. Callando

Obstara los daños de envidia e murmulos;

Y así navegando, los puertos seguros

Atrás quedan todos ya, cuanto más ando.

Si bien discernís mi limpio motivo,

[...]

Fuente: La Celestina, Fernando de Rojas, 1499, Editorial Porrúa [6]

Se puede observar cómo el nombre del autor aparece oculto en este pequeño fragmento de la obra, algo que sería complicado de apreciar sin prestar atención a ello.

La esteganografía, si bien está estrechamente relacionada con la criptografía, se distingue de esta última en que no altera el texto claro, sino que lo oculta. Sin embargo, es fundamental para comprender la historia de la criptografía ya que ha tomado un papel crucial en el desarrollo de la disciplina.

2.2. Criptografía

La criptografía es el estudio y práctica de técnicas que se utilizan para asegurar la comunicación y proteger información de accesos no autorizados. Su estudio aborda el cifrado de la información, la autenticación y la integridad de los datos entre otros aspectos.

La criptografía implica la modificación del texto claro utilizando una serie de reglas para obtener un nuevo texto, denominado criptograma o texto cifrado. Estas reglas modifican la información del mensaje real, provocando que sea ininteligible para un tercero desconocedor de ellas y fácilmente recuperable para uno conocedor. Comúnmente se conoce como "cifrar" o "encriptar" a la técnica de modificación de dicho texto.

Se deriva de la combinación de las palabras griegas "kryptos" (secreto, oculto) y "graphos" (escritura). De este modo, la criptografía posibilita el intercambio de mensajes, volviéndolos ilegibles sin ocultar su existencia. Su objetivo principal radica en asegurar que la información transmitida permanezca confidencial, íntegra y auténtica. La confidencialidad garantiza que solo las personas autorizadas tengan acceso a la información, la integridad asegura que la información no sea manipulada ni alterada, y la autenticidad previene la suplantación de identidad.

El proceso criptográfico es muy sencillo:

1. El emisor del mensaje toma la información a transmitir (texto claro) y la encripta.
2. El mensaje cifrado se envía al destinatario e idealmente ningún tercero podrá conocer el texto claro al no conocer el método de encriptación.
3. El receptor del mensaje es capaz de comprobar la integridad y autenticidad de los datos, desencriptar el mensaje y obtener el texto claro sin dificultades.

Anteriormente en el paso 2 se ha utilizado la palabra 'idealmente' porque siempre puede darse el caso de que un interceptor del mensaje cifrado sea capaz

Capítulo 2. Nociones básicas de criptografía

de recuperar la información original. Existe una disciplina cuyo objetivo es conocer la información original que el emisor transmite al receptor haciendo uso de diversas técnicas. Esta disciplina es conocida como criptoanálisis y ha desempeñado un papel fundamental en el desarrollo histórico de la criptografía tal como la conocemos, siendo una parte clave en la mayoría de los eventos históricos que se mencionarán en este trabajo. La combinación de la criptografía y el criptoanálisis se conoce como criptología.

Inicialmente podemos distinguir tres tipos de métodos criptográficos o de cifrado:

1. Transposición:

Este método cambia de lugar las letras o símbolos del mensaje para obtener uno o varios criptogramas, también conocidos como anagramas. Por ejemplo, la palabra RAMO puede ser encriptada como ARMO o CRIPTOGRAFIA como GRAFICOTRIPA, es importante eliminar las tildes y los espacios para evitar dar pistas a terceros. Para cada texto o palabra, el número de posibles criptogramas aparece representado por la fórmula de permutaciones con repetición:

a = Número de repeticiones de la letra a en el texto en claro.

b = Numero de repeticiones de la letra b en el texto en claro.

c = Número de repeticiones de la letra c en el texto en claro

...

$$\text{Criptogramas} = \frac{\text{Nº de letras!}}{a! \cdot b! \cdot c! \cdot \dots}$$

Por tanto, una palabra corta como RAMO tendrá

$$\text{Criptogramas} = \frac{4!}{1! \cdot 1! \cdot 1! \cdot 1!} = 24$$

(Las letras que no aparecen se omiten ya que $0! = 1$.)

Mientras que la palabra CRIPTOGRAFÍA tendrá

$$\text{Criptogramas} = \frac{12!}{2! \cdot 2! \cdot 2!} = 59875200$$

(Se omiten las letras que sólo se repiten una vez ya que $1! = 1$.)

Por tanto, se puede observar que cuánto más larga sea la palabra o texto, más seguro es el mensaje cifrado con el método de transposición para un tercero desconocedor de la técnica usada para la colocación de las letras.

2. Sustitución:

En lugar de sencillamente cambiar las letras de lugar, en este método se sustituye cada letra del texto claro por una o varias letras o símbolos. El caso en el que cada letra se sustituya por una única letra o símbolo se conoce como sustitución monoalfabética, mientras que si dicha letra puede sustituirse por dos o más letras o símbolos se conoce como sustitución polialfabética. El conjunto de símbolos o letras por el que se sustituye cada elemento del texto claro se conoce como alfabeto o diccionario. Un ejemplo de sustitución monoalfabética sería la siguiente:

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
<	>	.	-	,	^	+	'	;	¿	¡	\$	(	=	"	*		}	@	°	/	¬	#		€	&

Figura 2.2: Ejemplo de diccionario de sustitución monoalfabética.

En este caso, la palabra ramo se sustituiría por }<(".

3. Combinación de ambos:

También se pueden combinar los métodos de transposición y sustitución para dotar al mensaje o texto claro de una capa adicional de seguridad. Esta combinación no solo aumenta la complejidad del cifrado, sino que también aumenta el tiempo y los recursos necesarios para descifrarlo, creando un desafío mayor para los interceptores.

Es crucial en este punto comprender la distinción entre cifrar y codificar al hablar de criptografía, ya que ambos métodos han sido decisivos para el desarrollo histórico de esta disciplina y aunque comúnmente se traten como sinónimos, existe una diferencia de concepto entre ellos que radica en la manera de transformar el texto claro. Los códigos pueden considerarse un método criptográfico de sustitución, donde una palabra o frase se reemplaza con un símbolo, palabra o secuencia de números. Por otro lado, el cifrado es una de las técnicas fundamentales de la criptografía que consiste en convertir el texto en un formato ilegible mediante un algoritmo y una clave específica.

El procedimiento de cifrado se apoya en una clave ajena al texto claro, que juega un papel fundamental ya que para llegar de nuevo al texto claro a partir del texto cifrado es necesario conocer el procedimiento o algoritmo utilizado y dicha clave. Por otro lado, en la codificación se trata de escribir el texto claro en un formato específico para su transmisión o almacenamiento, no necesariamente implica ocultar información. Su objetivo principal es el almacenamiento y la transmisión de la información, para ello sustituye un texto por otro con base en el código utilizado, haciendo suficiente el conocimiento del diccionario o libro del mismo para descifrarlo. Esto hace que las cifras sean las específicas para la protección de la confidencialidad de los datos, llegando a permitir que los métodos de cifrado y descifrado se hayan convertido en públicos recayendo toda la seguridad en la clave utilizada. En 1883 el holandés Augusto Kerckhoffs

Capítulo 2. Nociones básicas de criptografía

von Nieuwenhof describió en uno de sus artículos para la revista *Cryptographie Militaire* el principio de Kerckhoffs, que establece que la seguridad de un sistema criptográfico no debe depender de mantener en secreto el algoritmo de cifrado, sino que debe depender de mantener tan solo en secreto la clave.

La clave es conceptualmente más amplia que un conjunto de letras o una palabra, puede ser una secuencia casi infinita de caracteres o una secuencia generada durante el proceso de cifrado. En este trabajo se establece que, en ausencia de un código con equivalencias, se considerará que se está ante un cifrado, incluso si no hay una clave específica. A medida que se avance en la historia se analizarán las ventajas y desventajas de ambos métodos para ocultar información.

Capítulo 3

Criptografía desde la Antigüedad hasta la Primera Guerra Mundial

La historia de la criptografía se inicia con la exploración de los métodos de ocultación de información más significativos, organizados cronológicamente desde la Antigüedad hasta la Primera Guerra Mundial. También se trata el campo del criptoanálisis, donde se refleja cómo a partir de la ruptura de la seguridad de ciertos métodos van surgiendo nuevos más seguros.

3.1. Primeros ejemplos

El primer ejemplo de criptografía del que se tiene constancia data de aproximadamente el 1900 a.C. Se trata de un grabado en piedra en la tumba del noble Khnumhotep II en el Antiguo Egipto. Se modificaron los símbolos habituales usados en la escritura jeroglífica con el objetivo de dotar al texto de una mayor importancia y hacerlo más decoroso. A pesar de que el propósito no era ocultar el mensaje, el texto original o texto claro se transformaba, lo que lo convierte en el caso más antiguo de criptografía según la bibliografía, aunque no cumpla con todas sus características inherentes.

En torno al 1500 a.C. aparecen de nuevo alteraciones de la escritura habitual en Mesopotamia. En este caso, se modificaban los signos cuneiformes para mantener en secreto la fórmula utilizada para fabricar barniz, lo que coincide con el objetivo de la criptografía tal y como la conocemos.

No sería hasta el siglo V a.C., concretamente en torno al año 430 a.C., en el que se dejarían por escrito algunos de los primeros casos de uso de métodos de comunicación segura. El artífice de esto fue el historiador griego Heródoto, considerado el padre de la historiografía gracias a "Historia", una magna obra que siglos después fue dividida en nueve libros. En ella se narra como Demarato (rey de Esparta, 515-491 a.C.) limpió la cera de una tablilla, escribió sobre la madera un mensaje de aviso a los griegos de los planes de invasión de Jerjes (gran rey persa, 519-465 a.C.), y volvió a colocar la cera pareciendo que la tablilla

Capítulo 3. Criptografía desde la Antigüedad hasta la Primera Guerra Mundial

estaba sin escribir. Esto provocó el fracaso de Jerjes en la invasión de Grecia viéndose envuelto en toda suerte de intrigas palaciegas. [7]

Otro ejemplo relata como Harpago, fiel servidor del rey Astiages, utilizaba liebres para comunicarse con Ciro, nieto de Astiages, quien fue condenado a muerte al nacer debido a un sueño del rey en el que su nieto tomaría su lugar. Harpago escondía el mensaje en la barriga de la liebre y le decía al mensajero que Ciro debía abrirla con sus propias manos. El mensaje revelaba el mal momento que vivía su abuelo tras perder la fidelidad de sus generales haciendo que Ciro lanzara un ataque y venciera.

En la obra se pueden encontrar más ejemplos de estos métodos esteganográficos, en los cuáles se oculta ingeniosamente el mensaje o texto en claro en lugar de modificarlo.

3.2. El método criptográfico de la escítala lacedemonia

Data del siglo V a.C y es considerado el primer sistema criptográfico militar de la historia. Fue utilizado por los lacedemonios (habitantes de Lacedemonia, región del sur del Peloponeso dominada por Esparta) durante las guerras del Peloponeso entre Esparta y Atenas. El sistema consistía en una vara de madera en la que se enrollaba un pergamino en espiral y se escribía el mensaje longitudinalmente. El pergamino se desenrollaba, las letras del texto claro quedaban distribuidas en otro orden y se enviaba a su destinatario, que solo sería capaz de descifrarlo si poseía una vara de madera del mismo diámetro convirtiéndolo en la clave del método. Cabe destacar que era importante eliminar los espacios durante la escritura con este sistema para dificultar aún más el conocimiento del texto claro para un tercero no deseado. He aquí un ejemplo visual:



Figura 3.1: Ejemplo de mensaje en escítala. Fuente: UGR [8]

Es posible abstraer el concepto de escítala y llevarlo a una matriz con tantas columnas como caras presente la escítala (letras que caben a lo largo de la misma) y tantas filas como letras se escriben en una vuelta o cara. Para obtener el mensaje encriptado es tan sencillo como transponer la matriz obtenida y leer el mensaje por filas, solo una persona que conozca las dimensiones de la matriz original (en el caso real, el diámetro de la vara para enrollar la escítala) podrá volver a rellenar dicha matriz, transponerla y recuperar dicho mensaje. En el ejemplo de la Figura 1 la matriz resultante al tener 8 caras y 4 letras en cada cara sería la siguiente:

3.3. Cifrado Atbash

S	E	N	D	M	O	R	E
T	R	O	O	P	S	T	O
S	O	U	T	H	E	R	N
F	L	A	N	K	A	N	D

Figura 3.2: Matriz de escítala para 3.1

El mensaje encriptado sería: “STSFEROLNOUADOTNMPHKOSEARTRNEOND”. Es importante eliminar los espacios a la hora de escribir y encriptar el texto claro para hacer más difícil el proceso de recuperación del mensaje.

Otro ejemplo para encriptar el mensaje “La felicidad está en el saber” en una escítala con 5 caras y 5 letras por cara sería:

l	a	f	e	l
i	c	i	d	a
d	e	s	t	a
e	n	e	l	s
a	b	e	r	

Figura 3.3: Matriz de escítala para ‘La felicidad esta en el saber’

Y el mensaje cifrado quedaría “lideaacenbfiseeedtlrlaas”. Dado que el criptograma contiene las mismas letras que el texto claro cambiadas de lugar, podemos determinar que la escítala lacedemonia es un método criptográfico de transposición. Este método supuso un cambio radical respecto a los métodos esteganográficos que se habían utilizado anteriormente.

3.3. Cifrado Atbash

El método criptográfico Atbash, también conocido como el método del espejo, aparece por primera vez en el Libro de Jeremías en el Antiguo Testamento y se utilizaba para no nombrar directamente a Babilonia. Se trata de un método de sustitución monoalfabética en el que cada letra se sustituye por aquella que está en la misma posición contando desde el final del alfabeto. Su nombre se inspira en el propio método: la primera letra del alfabeto hebreo, Aleph (A), es seguida por la última, Tav (T), ésta por la segunda, Beth (B), que, tras una A, es seguida por SH, proveniente de Shin (penúltima letra). Una forma análoga de llamar a este método es azby. Si se utilizan los números del 1 al 26 para codificar cada una de las letras del abecedario (sin la ñ) donde la A es 1, B es 2, C es 3, ..., Z es 26 entoces se puede determinar que:

$$\text{Nuevo N}^\circ \text{ letra} = 26 - \text{N}^\circ \text{ letra} + 1$$

Capítulo 3. Criptografía desde la Antigüedad hasta la Primera Guerra Mundial

La tabla de sustitución en el alfabeto utilizado a lo largo de este trabajo sería la siguiente:

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

Figura 3.4: Diccionario de sustitución Atbash

Un ejemplo con esta sustitución para “La felicidad está en el saber” sería “Oz uvorxrzwz vhgz vm vo hzyvi”, es evidente que sin conocer el método utilizado para la encriptación del texto claro es difícil hacerse con él a simple vista.

3.4. Eneas

Eneas el Táctico, escritor de la Antigua Grecia, es reconocido como un pionero por redactar el primer texto con instrucciones claras sobre cómo llevar a cabo comunicaciones seguras. Su obra estaba dedicada a la poliorcética, que es el arte del asedio, y en ella presentaba varias formas básicas de ocultación de mensajes en las comunicaciones. Uno de los trucos mencionados era el intercambiar todas las vocales de un texto por puntos, marcando con el número de puntos la vocal que se había eliminado, un ejemplo para el alfabeto latino y la oración “La felicidad está en el saber” sería:

L. f.l. . . c. . . d.d ..st. ..n ..l s.b..r

Otro método que propuso fue el uso de astragali, plural del hueso del tarso astrágalo también conocido como taba que ha dado lugar a multitud de juegos infantiles. La idea era perforar un agujero por cada letra del alfabeto, en el caso de este trabajo 26 agujeros, y utilizar un hilo para ir marcando cada una de las letras involucradas en el mensaje haciendo que vaya entrando y saliendo por los agujeros. Al desenredar el hilo, el receptor podría obtener el orden de los agujeros e ir sustituyendo por la letra que corresponde para obtener el mensaje. No es un método sencillo ni práctico para mensajes de cierta longitud, pero es una propuesta interesante. Estos métodos criptográficos son considerados inseguros a los ojos del siglo XXI. Sin embargo, representan un primer paso en el arte de la ocultación haciendo que para un analfabeto fuera indescifrable y para un alfabeto no sirviera con echar un vistazo rápido para comprender el texto.

Eneas también presentó métodos esteganográficos, uno de los cuales perduró hasta la Primera Guerra Mundial. Sugirió la técnica de tomar un documento con cualquier mensaje inocuo, por ejemplo un libro o un periódico, y marcar letras del texto de manera imperceptible, ya sea mediante agujeros o marcas sutiles, para encriptar el mensaje. Recomendó dispersar estas marcas a lo largo del texto para evitar despertar sospechas entre los lectores no destinatarios del mensaje. Más adelante se mostrarán aplicaciones de este método durante la Primera Guerra Mundial explicando su relevancia.

3.5. Polibio

Polibio (200-118 a.C.) fue considerado uno de los historiadores griegos más importantes al ser el primero en escribir una historia universal. Además, desempeñó un papel significativo en el desarrollo de la criptografía gracias a la tabla de Polibio. Defendió que para poder encriptar un texto se debían colocar todas las letras en una tabla con filas y columnas numeradas, normalmente cinco filas y cinco columnas con veinticinco posiciones, ya que sería una base suficiente para la comprensión. A partir de esta tabla, cada letra podía ser identificada mediante su fila y columna correspondiente, lo que permitiría cifrar el texto sustituyendo cada letra por la pareja de números que representaban su posición en la tabla. He aquí un ejemplo de la tabla de Polibio:

	1	2	3	4	5
1	A	B	C	D	E
2	F	G	H	I, J	K
3	L	M	N	O	P
4	Q	R	S	T	U
5	V	W	X	Y	Z

Figura 3.5: Tabla de Polibio

Nota: Se ha incluido en la misma celda la letra i y la j ya que solo se dispone de espacio para 25 letras, el receptor será capaz de interpretar el mensaje y saber qué letra es la correcta. Por ejemplo, en el caso de las palabras mirar y mjrar, es fácil reconocer el auténtico texto en claro.

Un ejemplo de encriptación con este método para la frase “La felicidad está en el saber” sería:

31 11 21 15 31 24 13 24 14 11 14 15 43 44 11 15 33 15 31 43 11 12 15 42

Se puede observar que se trata de un método criptográfico de sustitución mono-alfabética (cada letra tiene una única sustitución).

Una ventaja adicional de este método es la facilidad para transmitir el mensaje a distancia sin necesidad de un mensajero. Polibio sugirió usar antorchas, basándose en el sistema propuesto por Cleoxeno y Democleto en Alejandría para transmitir el texto claro. El número de antorchas en una mano indicaría el número de fila y el número en la otra mano el de columna. Esta idea de pasar letras a números ha sido fundamental en la historia de la criptografía, sirviendo para realizar operaciones matemáticas en los textos y dando origen a sistemas más seguros. Por tanto, Polibio sentó las bases sobre las cuales se desarrollaron numerosos sistemas criptográficos de la posteridad.

Se puede llegar a pensar que el mensaje a transmitir es sencillamente descifrable debido al uso del orden alfabético para colocar las letras en la tabla, facilitando a terceros no deseados la interceptación. Por ello, la tabla puede generalizarse

Capítulo 3. Criptografía desde la Antigüedad hasta la Primera Guerra Mundial

pensando una palabra como clave, escribiendo sus letras al principio de la tabla y, a continuación, escribiendo las restantes del alfabeto. Se muestra un ejemplo utilizando como clave la palabra SABER:

	1	2	3	4	5
1	S	A	B	E	R
2	C	D	F	G	H
3	I, J	K	L	M	N
4	O	P	Q	T	U
5	V	W	X	Y	Z

Figura 3.6: Tabla de Polibio con clave saber

Con esta nueva tabla, dos personas que conozcan la palabra clave son capaces de comunicarse, creando una dificultad mayor para un tercero no deseado de interceptar el mensaje.

3.6. Julio César

Julio César (100-44 a.C) fue uno de los políticos y militares más importantes de la Antigua Roma cambiando la historia de la humanidad de forma significativa en numerosos aspectos. En su obra “Comentarios sobre la guerra de las Galias” (De bello Gallico), describe su gran proyecto militar de conquista de la Galia, abordando también la forma de enviar mensajes de manera segura. Narra cómo, durante el asedio, se comunicaba con Cicerón (político romano) enviando mensajes escritos en griego, permitiendo la comprensión por parte de Cicerón y la indiferencia de las tropas enemigas ante el desconocimiento del idioma. No se sabe a ciencia cierta si escribió el mensaje usando el griego, o si escribió primero el mensaje en latín y más tarde sustituyó cada letra por su equivalente en griego. Este aspecto es de gran relevancia en el ámbito criptográfico, ya que implica un cambio de idioma o un cambio de alfabeto, técnicas utilizadas en la posteridad para garantizar la seguridad en la transmisión de mensajes.

Pero, sin ningún tipo de duda, el método criptográfico más importante que ha llegado hasta la actualidad es el cifrado César. Se trata de un método de sustitución monoalfabética que intercambia cada letra del mensaje original por la tercera letra que le sigue en el alfabeto. El diccionario de sustitución es el siguiente:

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
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

Figura 3.7: Diccionario cifrado César

El mensaje “La felicidad está en el saber” se sustituiría por “Od iholflgdg hvwd hq ho vdehu”, haciendo difícil su descryptación sin conocer el método que se ha utilizado para cifrarlo.

A lo largo de la historia se ha ido generalizando el método utilizando como clave el desplazamiento que se produce en el alfabeto. Si se utilizan los números del 0 al 25 para codificar cada una de las letras del abecedario (sin la ñ) donde la A es 0, B es 1, C es 2, . . . , Z es 25 entonces se puede determinar que:

$$\text{Nuevo N}^{\circ} \text{ letra} = \text{N}^{\circ} \text{ letra} + \text{clave (mód 26)}$$

De esta forma, por abuso de lenguaje, se conoce por cifrado César a todos aquellos sistemas que empleen un método de sustitución con una regla o clave algebraica similar a la utilizada por Julio César. Algunos ejemplos son los sistemas cuyas reglas fueran sustituir cada letra por la n -ésima siguiente, aplicando la fórmula mencionada anteriormente, o sustituir cada letra en posición par por la n -ésima siguiente y cada letra en posición impar por la n -ésima anterior, siendo n un número menor que la longitud del alfabeto.

3.7. Criptoanálisis árabe

Tras el fallecimiento de Mahoma en el año 632 d.C., la península Arábiga se encontraba bajo dominio islámico. No obstante, a lo largo del siglo siguiente, las regiones meridionales de Asia Central, la península Ibérica y el norte de África estaban bajo el liderazgo de gobernantes islámicos. Este rápido crecimiento generó una época de prosperidad que favoreció los avances en múltiples disciplinas. En el contexto de este trabajo, el empleo de codificaciones y cifrados se hizo habitual para garantizar una comunicación segura, llevando a los funcionarios a utilizarlos para proteger archivos y notas que consideraban confidenciales. Los árabes hacían uso de la sustitución monoalfabética en la que a cada letra le correspondía una letra o símbolo, algo que a estas alturas de la historia no suponía una gran innovación en el mundo de la criptografía respecto a los métodos anteriores.

Por el contrario, los árabes fueron pioneros en el campo del criptoanálisis, disciplina dedicada a obtener el texto en claro a partir de la interceptación de un mensaje cifrado. En este punto de la historia nace la “competición” en la que los criptógrafos se enfrentan a los criptoanalistas, quienes emplean métodos basados en el conocimiento y la lógica para romper los textos codificados o cifrados.

Se debe tener en cuenta que hasta ese momento la habilidad de leer y escribir no estaba demasiado extendida y las principales técnicas que se utilizaban para tratar de obtener el texto claro a partir del texto encriptado eran las siguientes:

1. Fuerza bruta

En los cifrados por sustitución de tipo César (métodos de sustitución monoalfabética que utilizaban el desplazamiento en un alfabeto como clave), se probaban todas las posibles claves hasta encontrar un mensaje con sentido. Un ejemplo de ataque por este método para la palabra “oleur” sería el

Capítulo 3. Criptografía desde la Antigüedad hasta la Primera Guerra Mundial

siguiente:

TEXTO CLARO	CLAVE
nkdtq	1
mjcsp	2
libro	3
khaqn	4
jgzpm	5
ifyol	6
hexnk	7
gdwmj	8
fcvli	9
ebukh	10
datjg	11
czsif	12
byrhe	13
axqgd	14
zwpfc	15
yvoeb	16
xunda	17
wtmcz	18
vslby	19
urkax	20
tqjzw	21
spiyv	22
rohxu	23
qngwt	24
pmfvs	25

Figura 3.8: Ataque por fuerza bruta para "oleur"

En 3.8, es evidente que la palabra libro es la única con sentido, permitiendo descifrar el mensaje encriptado. Si la longitud del mensaje y las letras del alfabeto utilizado no eran muy elevadas, en cuestión de horas el mensaje podía ser descifrado. En este caso, romper el sistema es sencillo ya que sabemos que cada letra se sustituye por otra que ocupa una determinada posición en el alfabeto pero conociendo solo que el alfabeto tiene 26 letras y se ha utilizado un método de sustitución este método no es válido.

2. Máxima verosimilitud

En algunas ocasiones se hacía uso de la posible información contenida en el mensaje original, del tema del mensaje a descifrar o de la estructura

del idioma utilizado. En el criptograma “Od iholflgdg hvwd hq ho vdehu” se puede deducir que la primera palabra “Od” es muy probable que sea un artículo que acompaña a un sustantivo y las palabras “hq ho” una preposición y un artículo, teniéndose en cuenta su longitud y el ser una estructura muy utilizada en el castellano. Si se deduce que el artículo inicial es “El”, entonces la letra E se sustituye por la letra O y se tendría un mensaje cifrado con clave 10, pero la estructura “hq ho” resulta en “xg xe” por lo que no es una clave válida. Por otro lado, si se tiene en cuenta que “Od” se sustituye por “La”, entonces la letra L se sustituye por la letra O y se tendría un mensaje cifrado con clave 3 en el que la estructura “hq ho” resulta en “en el” verificando su veracidad (el texto claro es “La felicidad está en el saber”).

Este método también podía ser utilizado para criptogramas encriptados haciendo uso de métodos de sustitución monoalfabética que no utilizaran desplazamiento. Sus principales desventajas eran que no era válido para mensajes de poca longitud y se convertía en un proceso muy tedioso si no utilizaban un cifrado de tipo César.

Todas las revelaciones que Mahoma había estado transmitiendo estaban recogidas en el Corán, pero no se conocía el momento exacto en el que habían sido reveladas. Ante esta incertidumbre, algunos teólogos tuvieron la idea de estudiar las palabras presentes en los textos con el objetivo de encontrar una evolución, ya que las expresiones de cualquier persona se van modificando con el transcurso del tiempo. Gracias al acceso que poseían dichos teólogos a otros textos de Mahoma con las formas de expresarse y las palabras más comunes a lo largo de cada etapa, llegaron a la conclusión de que se podía ordenar el Corán. Además de examinar las palabras y expresiones, dieron un paso más allá al realizar un análisis de las letras. Concluyeron que la disparidad en la frecuencia de aparición de las letras en los textos era tan significativa que podría ser relevante.

En definitiva, el estudio del Corán condujo a un descubrimiento clave en el ámbito de la criptografía: el análisis de frecuencias de las letras. La primera fuente en la que se describe esta técnica data del siglo IX y es el “tratado sobre el desciframiento de mensajes criptográficos”, obra de Al-Kindi, director de la Casa de la Sabiduría que se interesó por la filosofía, la astrología y las matemáticas. Se sostiene que el interés de Al-Kindi por la ruptura de códigos y cifras se debía a que algunos de los textos que deseaba leer o estudiar estaban encriptados y buscaba encontrar una puerta de acceso a esos textos claros.

La técnica consiste en examinar todas las palabras que hay en el diccionario de un determinado idioma y observar cuáles son las letras que se repiten con mayor frecuencia. Luego, se lleva a cabo el mismo estudio con las letras o símbolos del texto cifrado, sustituyendo unas por otras basándose en las equivalencias en la frecuencia de aparición. A medida que aumenta la longitud del texto encriptado, la precisión y fiabilidad del cálculo de frecuencias aumentan, lo que hace que el proceso sea más efectivo. En el castellano se puede determinar que la frecuencia de cada letra es la siguiente:

Capítulo 3. Criptografía desde la Antigüedad hasta la Primera Guerra Mundial

A	11,96%
B	0,92%
C	2,92%
D	6,87%
E	16,78%
F	0,52%
G	0,73%
H	0,89%
I	4,15%
J	0,30%
K	0,01%
L	8,37%
M	2,12%
N	7,01%
Ñ	0,29%
O	8,69%
P	2,77%
Q	1,53%
R	4,94%
S	7,88%
T	3,31%
U	4,80%
V	0,39%
W	0,01%
X	0,06%
Y	1,54%
Z	0,15%

Figura 3.9: Frecuencia de letras castellano. Fuente: Matemáticos, espías y piratas informáticos, National Geographic [3]

Para descryptar cualquier texto cifrado en el idioma español, bastaría con hallar las frecuencias de las letras o símbolos del texto a descifrar, compararlas con las frecuencias dadas en 3.9 y sustituir aquellas con frecuencias similares.

Cinco siglos más tarde llegó el análisis de frecuencia al occidente de Europa como método de ruptura de las cifras y los códigos. Esta técnica sencilla y accesible de origen árabe sentó las bases que siguieron un gran número de criptoanálisis para cambiar el transcurso de la historia. Más adelante se verán algunos ejemplos en eventos históricos de gran importancia.

3.8. Edad Media

Durante gran parte de la Edad Media no hubo avances significativos en el ámbito de la criptografía ya que los métodos no diferían mucho de lo que se había

hecho hasta el momento. El auge del poder eclesiástico condujo a que las figuras religiosas fueran las principales usuarias de la criptografía.

El papa Silvestre II, conocido como el papa matemático durante su pontificado entre el 999 y 1003, sustituyó los números romanos utilizados hasta ese momento por los números árabes que se utilizan a día de hoy. Supuso un hito en la historia de las matemáticas facilitando ciertas operaciones y cálculos, lo que permitió a individuos sin un conocimiento profundo llevarlos a cabo. Dentro de la criptografía, Silvestre II utilizaba las denominadas notas tironianas, un sistema taquigráfico inspirado en el romano Marco Tulio Tirón, secretario de Marco Tulio Cicerón. El sistema consistía en sustituir sílabas por símbolos o signos, algo muy parecido a los métodos de sustitución mencionados anteriormente.

Al igual que el estudio de los textos religiosos llevó a la cultura árabe a interesarse por el mundo de la criptografía, los teólogos cristianos se adentraron en él tras el estudio del Antiguo Testamento, en el que aparecía el método criptográfico Atbash. Roger Bacon fue el autor de “Epístola sobre las obras de arte secretas y la nulidad de la magia” en el siglo XIII en la que describía siete métodos de ocultación de mensajes entre los que se incluía la eliminación de vocales. Geoffrey Chaucer, autor de “Los cuentos de Canterbury”, también dejó pasajes breves encriptados con un alfabeto de símbolos.

3.9. Leon Battista Alberti

Leon Battista Alberti nació en Génova en 1404 y fue arquitecto, músico, poeta, matemático y filósofo. Trabajó para tres papas y la necesidad de cifrar mensajes de manera general y con seguridad le hicieron adentrarse en el mundo de la criptografía. Llegó a la conclusión de que los métodos de sustitución utilizados hasta la época eran débiles, debido a que para cada letra había un único sustituto constante y fijo, lo que facilitaba la descryptación por parte de un criptoanalista que empleara el análisis de frecuencia. Tras una reflexión, concluyó en su obra “Modus scribendi in ziferas” que la solución era utilizar varios alfabetos para llevar a cabo una sustitución según una regla y, después de un determinado número de sustituciones, cambiar de regla. Un ejemplo de este sistema con dos alfabetos sería el siguiente:

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

Figura 3.10: Diccionario con dos reglas de sustitución

Para encriptar el mensaje “La felicidad esta en el saber” utilizando 3.10 se sustituyen las letras en posición impar por su equivalente en la primera fila y las letras en posición par por su equivalente en la segunda fila, el resultado será: “Ri qcruhumim cnvd cz cr zdgll”. Como es posible observar, el análisis de frecuencia no es válido para obtener el texto claro ya que en unas ocasiones la letra A se sustituye por la D y en otras por la I. Este es solo un ejemplo de regla, otro

Capítulo 3. Criptografía desde la Antigüedad hasta la Primera Guerra Mundial

ejemplo sería sustituir por el equivalente en la primera fila la primera mitad de letras y por el equivalente en la segunda la otra mitad. No solo basta con que el receptor conozca el abecedario, también debe conocer las reglas que debe seguir.

Este tipo de sistema criptográfico se conoce como sustitución polialfabética, del que Alberti es considerado el creador. La propuesta que más peso ha tenido en la historia y que resistió durante siglos es el disco de Alberti, un dispositivo compuesto por dos anillos concéntricos. El anillo interior, fijo en el centro, presenta las letras minúsculas desordenadas y el símbolo "&" mientras que el anillo exterior lo rodea y contiene 20 letras ordenadas alfabéticamente y los números del 1 al 4.



Figura 3.11: Disco de Alberti. Fuente: Historia de la criptografía, Manuel J. Prieto [1]

Al girar el disco, cada una de las 24 posiciones del disco externo podía enfrentarse con cualquiera de las posiciones del disco interno, haciendo que el número de sustituciones para cada letra sea de 24. Para establecer una comunicación segura mediante este método, tanto el emisor como el receptor deben poseer un disco de Alberti con el disco interno reordenado de manera idéntica. En el proce-

so de cifrado, el emisor comienza el mensaje colocando dos letras en mayúscula que deben estar enfrentadas en el disco, y luego sustituye las letras del anillo externo por sus equivalentes en el anillo interno. Si el emisor incluye una letra mayúscula en el texto cifrado, indica que se debe realizar una recolocación de los anillos tomando esa letra como referencia.

Aunque este método permitía el análisis de frecuencia por tramos entre mayúsculas, la idea fue muy poderosa para la época ya que aportaba una mayor seguridad respecto a los métodos anteriores. Este tipo de sistema llevaría a cabo profundas modificaciones, haciendo que autores posteriores como Cardano o Tritemio lo llevaran a su máximo esplendor.

3.10. Nomenclátor

Durante los primeros años del Renacimiento, ante el auge del análisis de frecuencia, se buscaron métodos sencillos y efectivos de cifrado que consistían en sustituir las letras más frecuentes de forma alternativa por varios símbolos o letras, provocando que las frecuencias no fueran coincidentes. Este tipo de sustitución en el que las letras más comunes corresponden a varias letras o símbolos en el cifrado proporcionales a la frecuencia de la letra se conoce como sustitución homofónica.

La primera instancia documentada de este tipo de sustitución data de 1401, en la correspondencia entre Simeone de Crema y el duque de Mantua. Durante este intercambio, ambos acordaron las equivalencias de letras para generar el texto codificado, añadiendo algunos números y símbolos ajenos al alfabeto, con el fin de aumentar el número de elementos utilizados. La inclusión de estos símbolos adicionales permitía asignar varias sustituciones a las letras más comunes, redistribuyendo las frecuencias altas en algunas más bajas para equilibrarlas al resto.

A pesar de representar un avance considerable y de los esfuerzos por superar a los criptoanalistas, estos últimos elaboraban hipótesis y procedían a descubrir el texto cifrado mediante la prueba de diferentes opciones. En el caso de la correspondencia mencionada, en la que la palabra "Mantua" fue utilizada, la letra "A", al ser una de las letras más frecuentes, se habría sustituido por dos símbolos distintos, mientras que el resto de las letras permanecían sin cambio. Por lo tanto, bastaba con identificar patrones en los que las posiciones 2 y 6 de una palabra cambiaran para deducir las equivalencias y, así, descifrar el texto en claro.

Para evitar problemas y vencer a los criptoanalistas, se desarrollaron los nomenclátors durante el siglo XVI. Se trata de un concepto planteado por el secretario del papa Clemente VII, Gabriel de Lavinde, similar al de un diccionario que además del alfabeto cifrado, incluía una serie de palabras comunes codificadas. En ocasiones lo único que se utilizaba era un nomenclátor en el que se sustituían las palabras o expresiones más importantes sin modificar el resto del texto.

3.11. Johannes Trithemius

inicial de Leon Battista Alberti, sino que empleó un número de alfabetos igual al número de letras en el alfabeto original, lo que aumentó la seguridad del cifrado. Un ejemplo de esta tabla, conocida como tabula de Tritemio, con el alfabeto utilizado a lo largo de este trabajo es el siguiente:

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

Figura 3.13: Tabula de Tritemio

El proceso de encriptación con el cifrado de Tritemio a partir de 3.13 es relativamente simple. Se procede de la siguiente manera: la primera letra del texto claro se toma sin modificar; luego, se selecciona la fila cuya primera letra corresponde a la segunda letra del texto claro, y se toma la segunda posición de esa fila. Posteriormente, se busca el alfabeto que comienza con la tercera letra del texto claro y se selecciona la tercera posición, y así sucesivamente, hasta encriptar todo el mensaje. Por ejemplo, con el texto claro “La felicidad” se seguiría el siguiente proceso:

Se mantiene la letra L por ser la inicial. Se localiza la fila cuya primera letra es la A y se selecciona la segunda posición, es decir, la B. Ahora se localiza la fila que comienza por F y se escoge la tercera letra, la H. Siguiendo con el resto de letras del texto claro el resultado es el siguiente: LB HHPNIPLJN.

Para desencriptar el mensaje cifrado con el método de Tritemio, el receptor debe

Capítulo 3. Criptografía desde la Antigüedad hasta la Primera Guerra Mundial

buscar la letra del mensaje cifrado en la columna correspondiente a la posición de esa letra, y luego tomar la primera letra de la fila en la que se encuentre. Aunque este método no proporciona un alto nivel de seguridad, representó un avance significativo en la evolución de los cifrados, siguiendo la línea de lo expuesto por Alberti.

3.12. Pavía Girolamo Cardano

Pavía Girolamo Cardano fue un matemático italiano con formación en leyes y en medicina. Destacó por sus avances en álgebra al estudiar sobre las soluciones de las ecuaciones de tercer y cuarto grado, así como en el campo de la probabilidad y la estadística. Su interés por estas dos últimas ramas le llevaron a ahondar más en el mundo de la criptografía llegando a escribir tratados con recopilaciones sobre el tema.

Una de sus contribuciones más destacadas en este ámbito se refiere a lo que se considera la primera autoclave de cifrado de la historia. Propuso trabajar con la tabla de Tritemio vista anteriormente y utilizar como clave el texto claro, volviendo a comenzar por la primera letra de la clave con cada nueva palabra del texto a cifrar. Por ejemplo, para el texto claro “LA FELICIDAD ESTÁ EN EL SABER”, la clave para saber que alfabeto utilizar en cada letra sería “LA LAFELICID LAFE LA LA LAFEL”. Al ser el propio texto claro el que determina la clave se generan numerosos problemas de seguridad, provocando que esta idea no condujera a ningún avance práctico.

No obstante, Cardano logró dejar huella en el mundo de la criptografía al idear un método esteganográfico conocido como “La rejilla de Cardano”. Se utiliza un papel o cartón con diferentes agujeros hechos de manera irregular, se coloca sobre un papel en blanco y el emisor escribe el texto en los distintos agujeros, por letras, sílabas o palabras en función del tamaño. Una vez escrito el mensaje, se retira la rejilla y se completa la página con un texto inocuo pero coherente. Un ejemplo de este sistema se presenta en la siguiente figura:

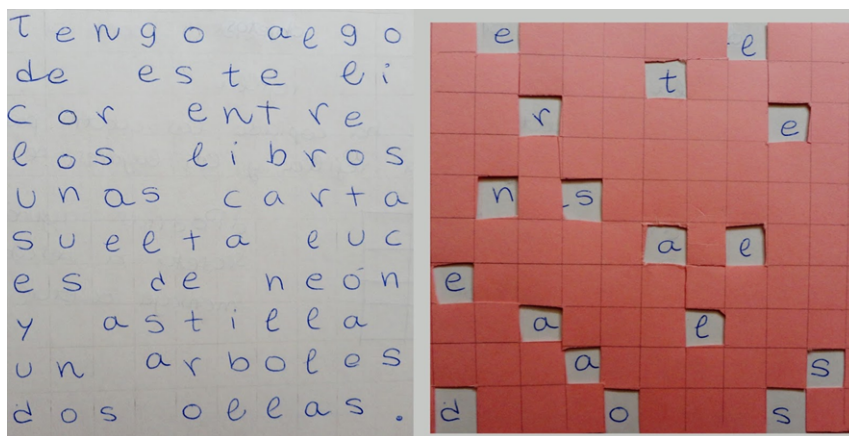


Figura 3.14: Ejemplo de rejilla de Cardano. Fuente: Matematiza tu realidad [10]

3.13. Giovan Battista Bellaso

Giovan Battista Bellaso fue un italiano renacentista del que se tiene constancia de sus avances en la sustitución polialfabética y el uso de claves. Siguiendo los conceptos de Tritemio, desarrolló un método para crear autoclaves más sofisticadas que las contempladas por Cardano. Su propuesta consistió en seleccionar múltiples palabras en latín u otro idioma, las cuales denominaría clave, y el texto a cifrar, escribiendo ambas en el papel de manera que cada letra de la clave estuviera alineada con cada letra del texto claro. La clave debería de ser repetida tantas veces hasta completar la longitud del texto a cifrar. Esto, junto con varios alfabetos identificados por su letra de cabecera, eran lo necesario. Un ejemplo de este tipo de cifrado con la clave “CLAVE”, el texto “La felicidad esta en el saber” y la tabla de Tritemio 3.13 sería:

CL AVECLAVEC LAVE CL AV ECLAV

LA FELICIDAD ESTA EN EL SABER

Basta con empezar a buscar en el alfabeto de la C la sustitución de la letra L, la sustitución de la A en el alfabeto de la L, y así sucesivamente. El resultado es el siguiente:

NL FZPKNIXEF PSOE GY EG WCMEM

Toda la seguridad del método se depositaba en la clave, si se creía comprometida bastaba con modificarla para volver a hacer seguras las comunicaciones.

3.14. Giovanni Batista Della Porta

Giovanni Batista Della Porta nació en 1535 y fue uno de los pioneros de la sustitución polialfabética combinada con el concepto de clave. Tuvo interés por muchas disciplinas, entre las que se encuentran las matemáticas y la criptografía, dejando por escrito la mayoría de sus ideas e investigaciones.

Uno de los métodos criptográficos que dejó por escrito giraba en torno a la misma idea que Bellaso había propuesto años atrás. Della Porta propuso una matriz con varios alfabetos en los que cada uno estaba escrito en dos filas, que debían contener todas las letras pero con un orden impredecible. En estos alfabetos cada letra a cifrar se sustituía por la letra que tenía arriba o abajo, dependiendo de su posición en la fila superior o inferior. Un ejemplo es el siguiente:

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

Figura 3.15: Ejemplo de matriz de Della Porta

En el alfabeto de 3.15 la sustitución de la letra A es la L y la sustitución de la letra L es la A. La elección de los distintos alfabetos se llevaba a cabo de la misma forma que en el método de Bellaso: se escogía una clave acordada entre emisor y receptor, se alineaba con el texto claro y se iniciaba la sustitución a partir de los

Capítulo 3. Criptografía desde la Antigüedad hasta la Primera Guerra Mundial

alfabetos correspondientes. Della Porta también ideó un método esteganográfico que implicaba utilizar un huevo duro. El procedimiento consistía en escribir sobre la cáscara con tinta de alumbre y vinagre, lo que resultaba en que el mensaje quedara marcado en la clara del huevo.

3.15. Blaise de Vigenère

Blaise de Vigenère nació en 1523 y fue el que se llevó la fama final en la revolución de los cifradores polialfabéticos. Fue un diplomático francés, lo cual le confería un profundo conocimiento en criptografía y durante su estancia en Roma alrededor del año 1550, adquirió un entendimiento de los métodos polialfabéticos presentados por Alberti, Tritemio y Della Porta, entre otros. No fue hasta 1562, año en el que se empezó a dedicar plenamente al estudio criptográfico tras haber acumulado suficiente riqueza, cuando analizó y desarrolló detenidamente todos los trabajos anteriores para desarrollar el método que lleva su nombre.

El método de Vigenère recuperó la idea establecida por Bellaso de combinar 26 alfabetos e ir saltando entre filas con el cifrado de cada letra. El emisor y el receptor debían establecer los alfabetos a usar y el orden mediante el uso de una clave que se alinearía con el mensaje a cifrar, normalmente de menos longitud que el texto claro por lo que debía de repetirse de forma cíclica como en los métodos anteriores. Para cifrar cada letra, se empleaba el diccionario asociado a la letra de la clave alineada con la letra correspondiente, y se seleccionaba la posición que la letra a cifrar ocupaba en el alfabeto inicial dentro del diccionario respectivo. La diferencia respecto al método propuesto por Bellaso fue que las letras de los alfabetos no tenían por qué estar en el orden habitual, sino que podían seguir otras configuraciones acordadas entre emisor y receptor. Un ejemplo haciendo uso de la tabla de Tritemio 3.13, la clave “CLAVE” y el texto “La felicidad esta en el saber” es el siguiente:

CL AVECLAVEC LAVE CL AV ECLAV

LA FELICIDAD ESTA EN EL SABER

Se comienza buscando en el alfabeto correspondiente a la letra C (tercera fila de la tabla de Tritemio 3.13) la posición de la letra L en el primer alfabeto (primera fila de la tabla de Tritemio 3.13). Al ser la posición 11, la sustitución para la letra L será la letra N, que ocupa dicha posición en el alfabeto de la C. A continuación se procede con la búsqueda de la posición de la letra A en el alfabeto de la L, y así sucesivamente hasta completar el cifrado del texto claro. El resultado final es:

NL FZPKNIYEF PSOE GY EG WCMEM

Esto prueba que en el caso de hacer uso de la tabla de Tritemio, el mensaje cifrado es idéntico al del método de Bellaso.

Vigenère publicó en 1586 su obra “Traicté des chiffres”, la cual incluía el método anterior y a día de hoy es considerada uno de los pilares de la disciplina. A pesar del gran avance que suponía y de la imposibilidad de los criptoanalis-

tas de descifrar el mensaje con las técnicas conocidas hasta la época (análisis de frecuencia, máxima verosimilitud o ataque por fuerza bruta), la mayoría de gobiernos y diplomáticos siguieron utilizando nomenclátors o sistemas menos seguros durante el siglo XVI. El paso del tiempo hizo que estos avances se fueran extendiendo y se popularizaran los métodos de sustitución polialfabética con clave, ya que en el caso de una ruptura del método bastaba con modificar las claves para recuperar las comunicaciones seguras.

3.16. Cifrado Pigpen

La francmasonería es una institución filantrópica calificada como la sociedad secreta más extendida del mundo. Su creación se remonta al siglo XVIII, cuando los ingleses cultos perseguían estar en comunión con otros y debatir cuestiones filosóficas, religiosas y vitales en un entorno organizado.[11]

En el ámbito criptográfico destacaron por el método que utilizaban para sus comunicaciones secretas o dejar mensajes escritos en algunos lugares. Se basaron en los métodos de los templarios utilizados para fines comerciales encontrados en documentos manuscritos del siglo XIII en los que aparece el alfabeto templario. Se trata de una forma de sustitución monoalfabética en la que a cada letra le corresponde un dibujo proveniente de la posición en la que se colocan las letras dentro de una tabla o una cruz. El funcionamiento es el siguiente:

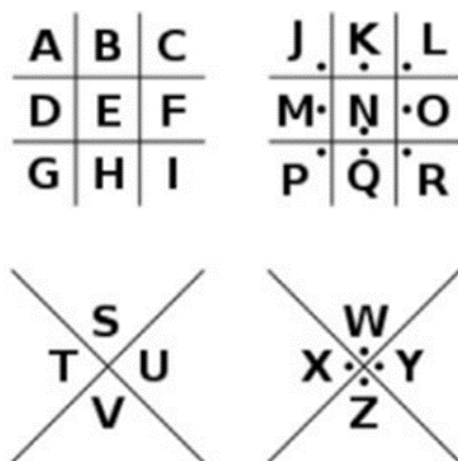


Figura 3.16: Cifrado Pigpen. Fuente: Historia de la criptografía, Manuel J.Prieto [1]

Este cifrado también se conoce como cifrado Pigpen o pocilga, y cada letra se describe por las líneas y puntos del lugar en el que se encuentra en 3.16, un ejemplo con el texto Plus Ultra es el siguiente:

PLUS ULTRA

7 6 5 4 3 2 1

A pesar de ser comúnmente atribuido a los francmasones, este método había sido utilizado previamente por los Tudor y lo usarían posteriormente grandes figuras como Napoleón. Fue descrito por Vigènere en “Traicté des chiffres”, que atribuyó su creación al oculista y teólogo Heinrich Cornelius Agrippa.

3.17. La Gran Cifra de París

Napoleón Bonaparte se autoproclamó emperador de Francia el 2 de diciembre de 1804 y dispuso de la inteligencia militar más avanzada que había tenido Francia a lo largo de su historia. Napoleón examinaba minuciosamente los periódicos extranjeros traducidos para conocer la situación y el movimiento de tropas que efectuaban sus contrincantes, con el objetivo de obtener una ventaja estratégica. Además, otorgó una gran importancia a la criptografía al cifrar toda su correspondencia para evitar la intromisión de espías extranjeros.

Hasta 1811, todos los generales de Napoleón empleaban métodos de sustitución monoalfabéticos, los cuales podían descifrarse mediante un análisis de frecuencia relativamente sencillo. Por consiguiente, se inició un proceso de sofisticación de los métodos criptográficos empleados para asegurar las comunicaciones y dificultar el trabajo de los criptoanalistas españoles y británicos. A principios de 1812, los franceses introdujeron un nuevo código conocido como la Gran Cifra de París con 1400 entradas que servían para sustituir letras, sílabas o palabras enteras. El código estaba basado en la Grand Chiffre de Luis XIV, uno de los ejemplos de nomenclátor más significativos dentro de la historia. Se repartieron los libros con las equivalencias a cada general para que fueran capaces de obtener los textos claros y comunicarse de forma segura al proporcionar un nivel de seguridad suficiente y ser más sencillo de usar que otros métodos.

Los franceses no eran cuidadosos con los envíos ya que confiaban plenamente en el poder de su cifra sin mostrar preocupación por aquellos mensajes que caían en manos enemigas. George Scovell, intendente que prestaba servicio en el ejército británico, aprovechó esta situación de indiferencia para hacerse con los mensajes encriptados y llevar a cabo un estudio exhaustivo. Empezó realizando un análisis de frecuencia que le llevó a descubrir los números que más veces se repetían y a relacionarlos con las letras o palabras más repetidas en el idioma francés obteniendo unos resultados iniciales prometedores. Muchos generales franceses combinaban el texto cifrado con texto claro, un error muy grave en la criptografía, haciendo que Scovell hiciera uso de análisis de máxima verosimilitud para buscar patrones gramaticales y encontrar nuevas sustituciones. Adicionalmente a estas dos técnicas, el tiempo, el uso y la gran cantidad de mensajes franceses de los que disponía el ejército británico provocaron la derrota del imperio francés contra España y Portugal en la Guerra de la Independencia. Este suceso es conocido comúnmente en el ámbito criptográfico como la derrota de

la Gran Cifra de París, constituyendo uno de los primeros hitos históricos donde el criptoanálisis adquirió una gran relevancia.

3.18. Julio Verne

Julio Verne, nacido en Nantes en 1828, fue uno de los autores más leídos de la historia. La narración de las aventuras de exploradores y los avances tecnológicos que plasmaba en sus obras, los cuales le otorgaron el aura de escritor visionario, fueron las principales fórmulas de su éxito.

En el mundo criptográfico Verne destacó por incluir numerosos métodos para ocultar información. En “Viaje al centro de la Tierra” la historia gira en torno a un pergamino con caracteres rúnicos que oculta un mensaje, lo que podía definirse como un código en el que cada carácter rúnico corresponde a una letra del abecedario. Pero, sin duda, su mayor aportación fue en su novela “Matías Sandorf”, publicada en 1885 por entregas, en la que hizo uso de una evolución de la rejilla de Cardano.

Se trata de una rejilla giratoria de tamaño $n \times n$ con agujeros como la que describió Cardano que al girar 90, 180 y 270 grados muestra diferentes secuencias de las letras que componen un mensaje. Para llevar a cabo una encriptación con esta rejilla primero se debía escribir el texto claro en una tabla cuadrada de n filas y n columnas (añadiendo letras si no es posible completar el cuadrado) y diseñar una rejilla $n \times n$ con aproximadamente $\frac{n^2}{4}$, es decir, una cuarta parte de agujeros que letras tiene la tabla, para poder cubrir todas en las 4 posiciones. Los agujeros han de distribuirse de forma que las posiciones que ocupan no coincidan con la de los agujeros al girar la rejilla 90, 180 y 270 grados, aquí se muestra un ejemplo para una tabla 6x6:



Figura 3.17: Ejemplo de rejilla de Julio Verne. Fuente: La criptografía, Luis Hernández Encinas [2]

Capítulo 3. Criptografía desde la Antigüedad hasta la Primera Guerra Mundial

Una vez se tiene la rejilla, se coloca sobre la tabla que contiene el texto claro y se escriben las primeras letras que quedan al descubierto (en este caso 9) en orden de aparición (de izquierda a derecha y de arriba a abajo). Se repite este proceso para cada uno de los giros obteniendo las 36 letras de la tabla, basta con que el receptor disponga de la misma rejilla y vaya escribiendo las letras en orden según la rejilla. Un ejemplo para el texto “Julio Verne fue el autor de Matías Sandorf” con la rejilla 3.17 es el siguiente:

- Primero se crea la tabla 6x6 con el texto añadiendo una letra al final, en este caso la W:

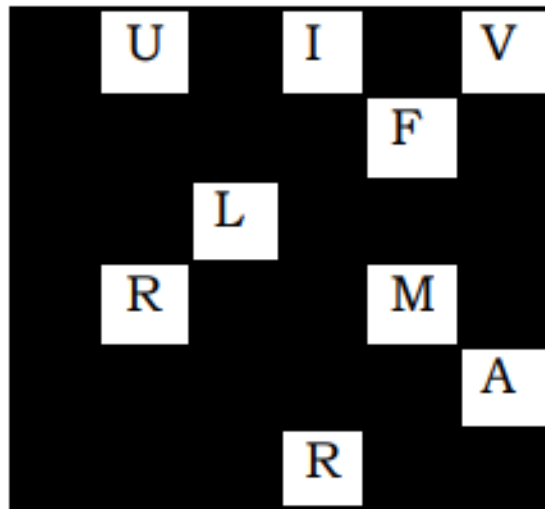
J	U	L	I	O	V
E	R	N	E	F	U
E	E	L	A	U	T
O	R	D	E	M	A
T	I	A	S	S	A
N	D	O	R	F	W

- Ahora se comienza colocando la rejilla 3.17 encima de la tabla para obtener las primeras 9 letras:

J				O	
	R		E		
E					T
		D			
T			S		

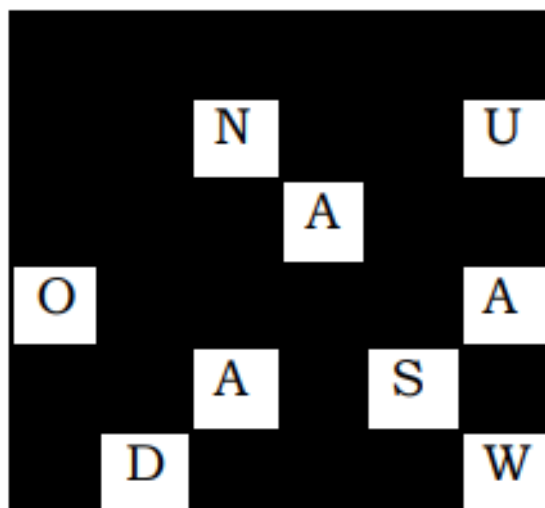
En este caso las primeras letras del mensaje cifrado serán: “JOREETDTS”.

- Se continúa con la rejilla 3.17 girada 90 grados en sentido horario:



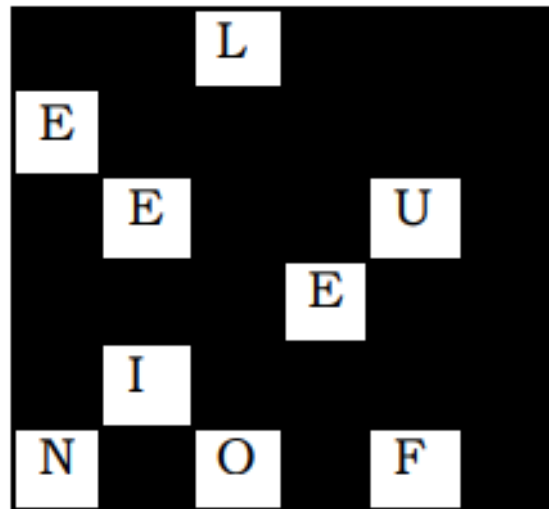
La nueva parte del mensaje cifrado es “UIVFLRMAR”.

- Al girar otros 90 grados:



Continúa con “NUAOAASDW”.

- Tras los últimos 90 grados:



Finalmente, se obtiene “LEEUEINOF”. El mensaje final es “JOREETDTSUIVFLR-MARNUAOAASDWLEEUEINOF”, basta con que el receptor posea la misma rejilla, se acuerde una posición inicial y vaya realizando los giros en el mismo orden para ir colocando las letras y recuperar el texto claro.

3.19. Lyon Playfair y Charles Wheatstone

Sir Charles Wheatstone fue un inventor y científico inglés nacido en 1802, diseñó una técnica de ocultación de mensajes que alcanzó cierto uso y popularidad a pesar de llevar el nombre de su amigo Lyon Playfair. Ambos compartían cierta afición por la criptografía intercambiándose mensajes cifrados y juntándose a menudo para descifrarlos y debatir sobre los mismos.

El método se inicia con la elección de una clave acordada de antemano por ambas partes, el emisor y el receptor, a través de un canal de comunicación seguro. Al igual que en el cifrado de Polibio, se escribe en una matriz 5x5 la clave, sin repetir ninguna letra, seguida del resto de letras del abecedario. Un ejemplo de matriz como el mostrado anteriormente en el cifrado de Polibio con la clave “SABIDURÍA” es el siguiente:

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

Figura 3.18: Tabla con clave ‘Saber’

3.19. Lyon Playfair y Charles Wheatstone

Se observa, de manera análoga al cifrado de Polibio, que la I y la J se incluyen en la misma casilla y no se repiten letras. Una vez establecida la tabla 3.18, el mensaje claro se divide en pares de letras, también conocidos como dígrafos, con la restricción de evitar la repetición de letras dentro de un mismo dígrafo. En situaciones donde el número de letras del mensaje sea impar o dos letras idénticas coincidan, se agrega una letra adicional entre las repetidas o al final que no altere el sentido de la frase, típicamente "X" o "W". A continuación se presenta un ejemplo de este procedimiento:

La felicidad esta en el saber

La-fe-li-ci-da-de-st-ae-ne-ls-ab-er

Para cada dígrafo hay tres opciones:

- Si ambas letras están en la misma fila de la matriz, se escogen las letras situadas a su derecha (si una de las letras está al final de la fila se escoge la primera de dicha fila).
- Si ambas letras están en la misma columna de la matriz, se escogen las letras situadas debajo (si una de las letras está al final de la columna se escoge la primera de dicha columna).
- Si las letras están en filas y columnas distintas se procede de la siguiente manera: se busca en la fila de la primera letra la letra que comparte columna con la segunda letra, y luego se repite el mismo proceso con la segunda letra.

Un ejemplo de este proceso con el dígrafo obtenido anteriormente se muestra a continuación.

- Se comienza con el dígrafo "LA", la posición de las letras en la tabla 3.18 es la siguiente:

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

Se trata del tercer caso, ya que no coincide la fila ni la columna. Para la letra A se busca en la primera fila la letra situada en la tercera columna, la I o la J (se tomará la I durante el proceso). Para sustituir la letra L se tomará la letra situada en la tercera fila y segunda columna, la H. La sustitución del dígrafo "LA" es "HI".

- Continuando con el dígrafo "FE", la posición de las letras en la tabla 3.18 es la siguiente:

Capítulo 3. Criptografía desde la Antigüedad hasta la Primera Guerra Mundial

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

Se trata del primer caso en el que ambas letras están en la misma fila. La letra F se sustituirá por la U, por ser la última letra de la fila, y la letra E se sustituirá por la F, por ser la situada inmediatamente a su derecha. La sustitución del dígrafo “FE” es “UF”.

- El siguiente dígrafo es “LI”, la posición de las letras en la tabla 3.18 es:

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

Se trata del segundo caso ya que ambas letras están situadas en la misma columna. La sustitución de la letra L será la Q y la sustitución de la letra I será la E. La sustitución del dígrafo “LI” es “QE”.

Se procede análogamente con el resto de dígrafos, el resultado final es “HI UF QE EB SB IF DN IR QU GI BI FC”.

El receptor del mensaje tendrá que llevar a cabo el proceso inverso para descifrar el mensaje, es decir:

- Si ambas letras del dígrafo cifrado están en la misma fila de la matriz, se escogen las letras situadas a su izquierda (si una de las letras está al principio de la fila se escoge la última de dicha fila).
- Si ambas letras están situadas en la misma columna de la matriz, se escogen las letras situadas inmediatamente encima (si una de las letras es la primera de la columna se escoge la última de dicha columna).
- Si las letras están en filas y columnas distintas se procede de la siguiente manera: se busca en la fila de la primera letra la letra que comparte columna con la segunda letra, y luego se repite el mismo proceso con la segunda letra.

3.19. Lyon Playfair y Charles Wheatstone

El método era sencillo de comprender y aplicar, ofreciendo una seguridad interesante siempre que la gestión de las claves fuera apropiada, dado que los digrafos estaban presentes. Con el paso del tiempo, el análisis de frecuencia se extendió a los pares de letras, lo que hizo que el sistema se volviera vulnerable. El estudio de los digramas requería una cantidad de texto mucho mayor para realizar un análisis concluyente, ya que en lugar de veintiséis letras posibles, como en un sistema de sustitución sencillo, los posibles digramas en un idioma son cientos (exactamente $26 \cdot 25 = 650$). El método Playfair es considerado una muestra paradigmática del avance en la criptografía al introducir sustituciones poligráficas, aquellas que en lugar de trabajar con letras trabajan con digramas, trigramas, ...

Otra de las contribuciones de Wheatstone en este ámbito fue un dispositivo muy similar al disco de Alberti. Consistía en dos alfabetos distribuidos a lo largo de dos circunferencias concéntricas, la más grande fija y la pequeña móvil, y dos agujas apuntando a cada alfabeto. Las agujas estaban unidas para que cuando se moviera la manecilla larga, la pequeña apuntara a la letra del cifrado que se debía tomar. Wheatstone incorporó un sistema por el que cada vez que se giraba completamente la manecilla larga, la posición de la corta variaba ligeramente no apuntando a las mismas letras para ofrecer una mayor seguridad. Este método se utilizó durante la Guerra Civil Española por el ejército de Franco.



Figura 3.19: Cifrador de Wheatstone. Fuente: Numerentur [12]

3.20. Babbage y Kasinski

Nacido en 1791, Charles Babbage fue hijo de un banquero, circunstancia que le otorgó la libertad financiera suficiente para dedicarse por completo a la resolución de problemas que desafiaban su mente, sin la obligación de buscar otras fuentes de ingreso. Fue un distinguido miembro de la Real Sociedad de Matemáticas londinense, de la Sociedad Astronómica y de la Sociedad Estadística, alcanzando numerosos logros durante su carrera. También destacó por su heterogeneidad en el mundo de la innovación.

Se reconoce a Babbage como el precursor de la computación gracias a su máquina diferencial, cuyo objetivo era obtener mecánicamente las tablas matemáticas para calcular logaritmos, funciones trigonométricas etc. En 1823, Charles presentó la primera versión de su máquina diferencial, financiada parcialmente por el gobierno británico, sin embargo, tanto en este intento inicial como en su segundo intento posterior, no logró alcanzar su objetivo deseado. A pesar de no haber logrado desarrollar el prototipo, su diseño supuso una revolución por el mero hecho de ser programable. Este diseño sentó las bases para la noción fundamental de contar con una memoria, un procesador y un conjunto de instrucciones, por ejemplo "si X, hacer Y", que ha sido fundamental en el desarrollo de la informática moderna. La criptografía y su estrecha relación con las matemáticas fueron un campo muy atractivo para el inglés, quien se interesó por el criptoanálisis desarrollando una habilidad excepcional para descifrar textos.

Recibiendo numerosas consultas y peticiones a lo largo de su carrera, en 1854, se enfrentó a un desafío particular cuando el dentista John Hall Brock Thwaites afirmó haber creado un cifrado revolucionario e indescifrable. Babbage identificó que el cifrado propuesto por el dentista era idéntico al cifrado de Vigenère, concebido tres siglos atrás. Sin embargo, el dentista se negó a admitirlo y desafió a Babbage a intentar romper su método.

Tras una investigación exhaustiva, Babbage identificó un problema relacionado con las repeticiones en el texto cifrado. Si la longitud del texto era lo suficientemente extensa, era posible detectar secuencias o palabras repetidas que facilitaban la revelación de la clave, comenzando por su longitud. Estas repeticiones podrían indicar que la misma palabra en el texto claro había sido codificada utilizando la misma parte de la clave. A medida que aumenta la longitud de las repeticiones, también aumenta la probabilidad de repetición tanto en el texto claro como en la clave, lo que proporciona información más útil. El método de Babbage permitía determinar la longitud de la clave y progresar en la búsqueda de las letras de la misma mediante un análisis adicional de la frecuencia de las apariciones. Este hallazgo tenía el potencial de revolucionar el campo de la criptografía en la época, sin embargo, el brillante matemático optó por mantenerlo en secreto. No fue hasta el siglo XX, tras su fallecimiento, que este descubrimiento salió a la luz gracias al estudio profundo de sus notas.

Para el momento en que el método de Babbage finalmente salió a la luz, el oficial del ejército prusiano retirado Friedrich Wilhelm Kasiski ya había logrado publicar en 1863 una técnica muy similar para romper este tipo de cifrados. Al

identificar palabras y secuencias repetidas, Kasiski dedujo que la distancia entre ellas era un múltiplo de la longitud de la clave. Por lo tanto, propuso encontrar el máximo común divisor de todas las distancias entre las repeticiones para determinar la longitud de la clave, que sería ese número o algún factor primo del mismo. Una vez que se descubría la longitud de la clave, el siguiente paso consistía en dividir el texto cifrado en bloques de acuerdo con esta longitud y luego llevar a cabo un análisis de frecuencia. Dado que estos cifrados seguían un enfoque similar al cifrado César, donde cada letra se desplaza un número fijo de posiciones en el alfabeto, este análisis de frecuencia permitía identificar patrones recurrentes y deducir las letras de la clave correspondientes a cada posición en el bloque.

Cuando Kasiski hizo público su método para la descryptación de cifrados polialfabéticos, los criptoanalistas recuperaron una posición de ventaja que no habían tenido desde la introducción de este tipo de métodos de cifrado. En este punto, los libros de códigos se convirtieron en la única opción sólida para mantener comunicaciones seguras.

3.21. Samuel Morse

A finales del siglo XVIII, tuvo lugar una revolución en el ámbito de las comunicaciones con la presentación, en 1792, del primer telégrafo óptico por parte del inventor francés Claude Chappe. Este telégrafo óptico consistía en un dispositivo que empleaba una serie de brazos articulados para establecer comunicaciones a distancia mediante señales visuales. Se trataba de una forma mejorada y avanzada de los sistemas de comunicaciones esteganográficos mencionados anteriormente como las antorchas o el humo.

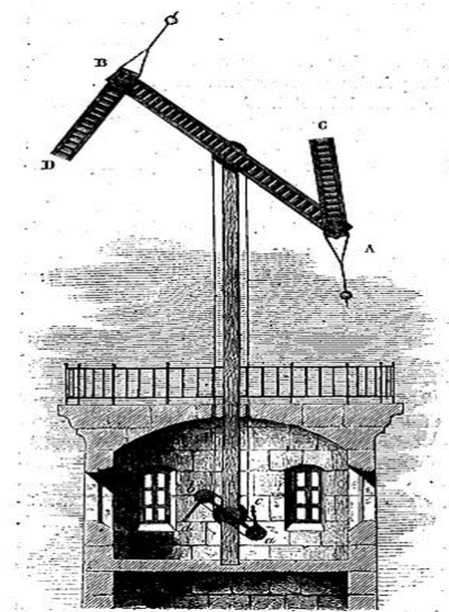


Figura 3.20: Telégrafo de Chappe. Fuente: ZTFNews [13]

Capítulo 3. Criptografía desde la Antigüedad hasta la Primera Guerra Mundial

Los telégrafos ópticos fueron utilizados en cierta medida a lo largo del siglo XIX, a pesar de depender en gran medida de las condiciones climáticas y de tener limitaciones en cuanto a su uso horario. La eficacia de la transmisión de mensajes estaba condicionada por la capacidad del receptor para visualizar el mensaje a distancia, lo que implicaba que fenómenos atmosféricos como la niebla o la lluvia, así como la falta de luz durante ciertas horas, obstaculizaban su propagación.

Los avances científicos, como la comprensión de la interacción entre las fuerzas eléctricas y magnéticas junto con el impulso generado por la revolución industrial, allanaron el camino para el desarrollo del telégrafo eléctrico. Este dispositivo revolucionario permitía enviar mensajes a larga distancia mediante la transmisión de impulsos eléctricos a lo largo de cables conductores, marcando así un hito en la historia de las comunicaciones.

El telégrafo eléctrico suprimió las limitaciones del telégrafo óptico al posibilitar la transmisión de mensajes en cualquier momento, independientemente de las condiciones climáticas. Con la transmisión de impulsos, surgió la necesidad de encontrar métodos efectivos para codificar los mensajes, asegurando así la comprensión entre el emisor y el receptor. Este proceso de codificación implicaba transformar la información que se quería transmitir en una forma sencilla que pudiera ser interpretada de manera precisa y confiable a través del canal eléctrico.

Samuel Morse fue el creador del código telegráfico más utilizado en las comunicaciones telegráficas. La principal razón de su éxito radica en su simplicidad: el código Morse solo requiere que el circuito eléctrico se cierre durante diferentes intervalos de tiempo para enviar pulsos más cortos o más largos, los cuales el receptor interpreta como puntos y rayas. Cada una de las letras del alfabeto, así como los números y otros caracteres, tiene su equivalente en una combinación específica de puntos y rayas. Esta codificación simple pero efectiva permitía enviar mensajes de cualquier longitud sin limitaciones, lo que lo convertía en una herramienta invaluable para la comunicación a larga distancia.

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	--..	0	-----	1	.-----
2	..----	3	...--	4	-	5	
6	-....	7	--...	8	---..	9	----.

Figura 3.21: Código Morse

El telégrafo carecía de medidas de seguridad, ya que cualquier individuo podía interceptar las líneas y acceder al contenido de los mensajes. Aunque los empleados de las empresas telegráficas se comprometían a preservar la confidencialidad de las comunicaciones, esta promesa no garantizaba la seguridad absoluta. Dado que la correspondencia entre cada letra del alfabeto y los puntos y rayas del código Morse era ampliamente conocida por la población, en la mayoría de los casos se empleaba un método criptográfico adicional para proteger la privacidad de las comunicaciones.

3.22. Auguste Kerckhoffs

A finales del siglo XIX, el lingüista holandés Auguste Kerckhoffs presentó los fundamentos teóricos que continúan siendo relevantes en la actualidad y que permiten definir los métodos criptográficos como seguros. Su idea principal sostiene que la seguridad de un sistema criptográfico radica en la clave, estableciendo así un modelo conceptual fundamental que subyace a la mayoría de los métodos criptográficos. A esta idea se la conoce como el principio de Kerckhoffs y se parte del supuesto de que el criptoanalista conoce el método utilizado y tiene acceso al mensaje cifrado.

Kerckhoffs marcó una serie de reglas como elementos deseables en cualquier método o sistema criptográfico:

1. El sistema debe ser irrompible en la práctica a pesar de que no lo sea teóricamente.
2. La efectividad de un sistema no debe depender de que permanezca en secreto.
3. La clave debe ser memorizable y no se debe recurrir a notas escritas.
4. Los criptogramas deben dar resultados alfanuméricos.
5. El sistema debe ser sencillo.
6. Una única persona debe ser capaz de operar el sistema.

Años después, Claude Shannon, considerado el padre de la teoría de la información, reinterpretó el principio de Kerckhoffs de manera más clara y directa: se debe asumir que el adversario conoce el sistema. Estas reglas transformaron la perspectiva de la criptografía, teniendo un impacto significativo en los eventos históricos del siglo XX.

3.23. La sala 40 y el telegrama Zimmermann

El 28 de julio de 1914, el Imperio austrohúngaro declaró la guerra a Serbia, marcando el inicio de la Primera Guerra Mundial. Antes de que Gran Bretaña se viera envuelta en el conflicto, tuvo lugar una conversación entre el responsable de inteligencia del Almirantazgo, Henry Francis Oliver, y el director de Educación Naval, sir James Alfred Ewing, con el objetivo de promover la creación de una

Capítulo 3. Criptografía desde la Antigüedad hasta la Primera Guerra Mundial

unidad de criptoanálisis. Oliver le expresó a Ewing, quien era un gran aficionado a la criptografía, que en su oficina había recibido comunicaciones interceptadas a los alemanes. Sin embargo, debido a que estaban cifradas o escritas en un código desconocido, los servicios de inteligencia no habían logrado descifrar los mensajes y obtener su contenido claro.

La entrada de Gran Bretaña en la guerra llevó a que Ewing aceptara el cargo de director del departamento de criptoanálisis del Almirantazgo de manera individual en sus inicios. Consciente de que sus conocimientos en criptoanálisis no eran lo suficientemente profundos, Ewing inició un periodo de estudio mientras continuaban interceptándose mensajes alemanes. El aumento del flujo en las comunicaciones alemanas generó la necesidad de reclutar a personas de confianza de la escuela naval, ya que en este tipo de tareas, la confidencialidad y la discreción son casi tan cruciales como los conocimientos técnicos. El despacho de Ewing, donde trabajaban, se volvía cada vez más pequeño a medida que se unían nuevos miembros, lo que llevó a la decisión de trasladarse a una sala de acceso restringido designada con el número asignado por el Almirantazgo: la sala 40 o Room 40.

En enero de 1917 la sala 40 interceptó un telegrama encriptado cuyo destinatario era el embajador alemán en Washington y el remitente Arthur Zimmermann, ministro de asuntos exteriores alemán. El telegrama estaba encriptado con el código 0075, recientemente adoptado por los alemanes y del cual los británicos tenían un conocimiento limitado, lo que dificultaba su análisis. Los criptoanalistas comenzaron a sospechar que los primeros grupos de números seguían un patrón común en la mayoría de las comunicaciones de este tipo: el primer número representaba el número de telegrama, el segundo la codificación, el tercero el emisor y los dos siguientes la fecha de envío. Además, observaron que la palabra "STOP" era frecuente en este tipo de mensajes, sugiriendo que podría ser el siguiente código a identificar. Aunque el código 0075 asignaba varios números a palabras comunes, la debilidad humana a menudo constituía una vulnerabilidad en la seguridad. Los operadores responsables de codificar los mensajes desarrollaban el hábito de memorizar ciertas secuencias para evitar consultar el libro de códigos con frecuencia, con el fin de reducir el tiempo dedicado a una tarea tan monótona y sistemática. Esta práctica resultó en la repetición de muchas secuencias, lo que facilitó la aplicación de técnicas como la máxima verosimilitud y el análisis de frecuencia para el criptoanálisis.

En las primeras horas tras su envío, el criptoanalista de la sala 40, Nigel de Grey, con la ayuda de su equipo, logró descifrar una parte del mensaje, obteniendo información suficiente que le permitió comprender el significativo impacto y las posibles consecuencias del texto alemán. Zimmermann proponía a México formar una alianza con el objetivo de invadir Estados Unidos y persuadir a Japón para que también declarara la guerra al gigante norteamericano. La sala 40 tenía la responsabilidad de verificar que el texto descifrado fuera preciso y que el emisor fuera el gobierno alemán antes de tomar decisiones precipitadas que pudieran afectar el curso de la guerra.

El telegrama llegó a Washington al día siguiente de ser enviado, donde la em-

bajada alemana obtuvo el texto claro y lo encriptó de nuevo utilizando el código 13040 antes de ser enviado a México. Un mes después, la sala 40 recibió el telegrama enviado a México y lograron obtener el texto claro, ya que este código había sido previamente descifrado y se consideraba menos seguro que el código 0075. El 22 de febrero de 1917, el telegrama fue entregado al embajador de Estados Unidos en Londres, lo que provocó la indignación del presidente Wilson y condujo a un acercamiento de Estados Unidos hacia la intervención en la guerra.

Este evento es ampliamente reconocido como uno de los más significativos en la historia de la criptografía debido al cambio radical que provocó en el curso de la guerra. La intervención de Estados Unidos en abril de 1917 aceleró la derrota de Alemania y el fin del conflicto. Los alemanes confiaron en la seguridad de un libro de códigos, pero no consideraron que, a pesar de las numerosas entradas disponibles, los usuarios tendían a utilizar solo una parte de ellas, lo que facilitó el trabajo de los criptoanalistas en la descryptación.

3.24. ADFGX

Tras el fracaso del telegrama de Zimmermann, el ejército alemán inició una búsqueda activa de métodos cifrados más seguros con el propósito de superar las limitaciones de los libros de códigos. Dado que el cifrado Vigenère se consideraba vulnerable, se convocó a un destacado conjunto de expertos en criptografía con la tarea de proponer sistemas de cifrado más robustos. Simultáneamente, un equipo de criptoanalistas se dedicó a la tarea de desentrañar estos métodos con el fin de identificar la opción óptima. De entre todas las cifras analizadas, se seleccionó la que había creado el coronel Fritz Nebel, denominada oficialmente como GEDEFU 18 (Geheimschrift der funkler 18, Cifra de telegrafistas 18) aunque pasó a la historia como ADFGX.

Al igual que Polibio hace varios siglos, Nebel propuso representar las letras del alfabeto en una tabla 5x5. Sin embargo, en lugar de utilizar números, optó por emplear las letras A, D, F, G y X. La elección de estas letras se basó en la simplicidad para ser codificadas en Morse dada la experiencia en telegrafía de Nebel: A(-.), D(-..), F(..-.), G(-.-) y X(-.-.-). A continuación se presenta un ejemplo de tabla utilizando la palabra "sabiduría" como clave:

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

Figura 3.22: Tabla ADFGX con clave 'sabiduría'

Capítulo 3. Criptografía desde la Antigüedad hasta la Primera Guerra Mundial

En el caso de 3.22, la palabra "felicidad" se cifra como "DX DG FG AG DF AG AX AD AX". Sin embargo, este tipo de cifrado era fácilmente descifrable mediante la técnica del análisis de frecuencia, ya que se trata de una sustitución mono-alfabética. Por ello, Nebel decidió incluir un segundo paso dentro del método involucrando una segunda clave.

Se debe elegir una nueva clave, en este caso se utiliza la palabra "clave", y colocar el texto restringiendo el número de letras por fila con el número de letras de la clave:

C L A V E

2 4 1 5 3

D X D G F

G A G D F

A G A X A

D A X

Los números "24153" indican el orden alfabético de las letras de la clave y el orden en el que se deben seleccionar las columnas para obtener el texto cifrado. En este caso, el mensaje cifrado final sería: "DGAXDGADFFAXAGAGDX". Se trataba de un método que combinaba técnicas de sustitución y transposición otorgando una mayor seguridad a las comunicaciones. Más adelante se añadió una nueva letra en el código (fila y columna) para introducir los dígitos del 0 al 9, por lo que pasó a conocerse como ADFGVX.

Los enemigos de los alemanes quedaron desconcertados al interceptar por primera vez mensajes con este tipo de cifrado, ya que era algo muy distinto a los códigos que venían utilizando hasta ahora. Este nuevo cifrado permitió a los alemanes avanzar posiciones hasta París mientras se preparaba una gran ofensiva, lo que obligó a los aliados a estudiar exhaustivamente dichos mensajes para anticiparse y resistir con mayor determinación.

Georges Jean Paivin, profesor y criptógrafo del ejército francés, llevó a cabo un estudio exhaustivo del sistema y desarrolló un método para desvelar las claves empleadas por los alemanes, permitiendo así el descifrado de sus mensajes. Este método, aunque efectivo, era costoso y demandaba considerables recursos, ya que implicaba la prueba de múltiples combinaciones para lograr la pronta obtención del texto claro. Esta estrategia resultó ser fundamental para prever las ofensivas alemanas, evitar la toma de París y revertir el avance enemigo en junio de 1918. La guerra culminó el 11 de noviembre de 1918 con Alemania firmando el Armisticio de Compiegne debido a la situación insostenible de sus fuerzas. Posteriormente, en junio de 1919, se firmó el Tratado de Versalles, en el cual Alemania fue obligada a indemnizar a las naciones afectadas por los daños ocasionados durante el conflicto.

Medio siglo después, Paivin y Nebel coincidieron y discutieron sobre la ruptura de la cifra ADFGX. Nebel explicó a Paivin que había propuesto una transposición adicional en el proceso, pero esta idea fue rechazada por las autoridades alemanas debido a las dificultades prácticas para implementarla. Paivin comentó que, probablemente, en ese caso no habría sido capaz de descifrar el código, lo que habría cambiado el curso de la guerra.

Capítulo 4

Criptografía en la Segunda Guerra Mundial

La Segunda Guerra Mundial, el conflicto bélico más catastrófico y significativo del Siglo XX, fue un punto de inflexión en la humanidad. Las tensiones políticas, económicas y territoriales gestadas desde el final de la Primera Guerra Mundial provocaron la inestabilidad y expansión de regímenes totalitarios como el Tercer Reich en Alemania o el fascismo en Italia.

Los errores cometidos en la Primera Guerra Mundial, junto con la popularidad creciente de la maquinaria tras la Revolución Industrial, propiciaron el auge de los dispositivos que combinaban comunicación y cifrado sin la necesidad de recurrir a un criptógrafo. El capítulo abarca los principales sistemas utilizados para el intercambio seguro de mensajes durante el conflicto, además del papel fundamental de las matemáticas en el criptoanálisis que contribuyeron a la victoria aliada en 1945.

4.1. Gilbert Vernam

Gilbert Vernam, ingeniero estadounidense trabajador en el laboratorio de AT&T en Manhattan, fue designado en 1917 para liderar un proyecto de investigación secreto cuyo propósito era la mejora de la seguridad de las comunicaciones a través de teletipos o telégrafos de impresión. La seguridad de estos sistemas era inexistente, dado que un oscilógrafo podía detectar las fluctuaciones de la corriente y decodificar la información transmitida, de manera similar a la lectura del código Morse en los telégrafos.

En el telégrafo de impresión o teletipo se empleaba un código similar al Morse, desarrollado por Émile Baudot en 1874. Cada letra se codificaba mediante cinco pulsos, análogos a los futuros bits, aunque en lugar de representar 0 o 1, correspondían a la presencia o ausencia de un pulso eléctrico. Las combinaciones posibles de estos cinco pulsos eran $2^5 = 32$, abarcando valores para todas las letras y caracteres adicionales como espacio, cambio a número o cambio de párrafo. En el teclado del emisor, se seleccionaban las letras a transmitir y se

Capítulo 4. Criptografía en la Segunda Guerra Mundial

generaban los pulsos correspondientes a través de la línea. Al llegar al destino, estos pulsos quedaban registrados en una cinta. Esta cinta se perforaba o no según la presencia de los pulsos eléctricos, cerrando un circuito cuando había una perforación y manteniéndolo abierto en caso contrario.

Vernam contempló la posibilidad de integrar los pulsos eléctricos generados por el teclado con otros pulsos externos, o la ausencia de los mismos. Estos nuevos pulsos, o su falta, funcionarían como una clave. Este proceso implicaría la sincronización de dos cintas, donde la combinación de perforaciones o su ausencia en cada una determinaría el valor del pulso final. En 1919 se patentó el cifrado Vernam que seguía la tabla 4.1:

Cinta de mensaje	Cinta de clave	Resultado
Agujero	Agujero	Ausencia
Agujero	Ausencia	Agujero
Ausencia	Agujero	Agujero
Ausencia	Ausencia	Ausencia

Cuadro 4.1: Tabla de combinaciones del cifrado Vernam

En el destino, la decodificación resulta sencilla si se dispone de una cinta de clave sincronizada con el mensaje cifrado. Como ejemplo, si tanto la clave como el texto cifrado presentan una perforación o agujero, el valor correspondiente en la cinta del texto plano debe ser una ausencia de pulso. Esta relación se refleja también en la tabla de verdad de la operación XOR, ampliamente reconocida en el ámbito computacional:

CIERTO+CIERTO = FALSO

CIERTO+FALSO = CIERTO

FALSO+CIERTO = CIERTO

FALSO+FALSO = CIERTO

El sistema Vernam destacaba por su facilidad, al incluir la cinta de la clave el propio sistema era el encargado de cifrar el mensaje antes de enviarlo y de descifrarlo en el destino si se introducía la misma cinta. Sin embargo, en los primeros prototipos, la cinta con la clave giraba de manera circular, lo que provocaba la repetición de la misma y volvía al sistema vulnerable al método de Babbage y Kasiski. Varios ingenieros de AT&T concluyeron que el sistema podría ser resistente al criptoanálisis si la clave fuera aleatoria e infinita, pero en la práctica, esas condiciones eran imposibles de cumplir.

Este sistema fue pionero de la idea de combinar en un único dispositivo comunicación y cifrado, sentando las bases del desarrollo de máquinas como los cifradores de rotación y la máquina Enigma.

4.2. Los cifradores de rotación

La idea de Gilbert Verman de combinar cifrado y comunicación en un mismo dispositivo había demostrado ser sumamente eficaz. La aparición de las máquinas de escribir a principios del siglo XX en oficinas e instalaciones gubernamentales había incitado a empresas e inventores a buscar métodos para desarrollar dispositivos similares a las máquinas de escribir que incluyeran el cifrado de forma interna. Esto evitaría los errores humanos surgidos durante el cifrado en la Primera Guerra Mundial.

Los inventores comenzaron a buscar inspiración en ideas previas de sistemas criptográficos para alcanzar su objetivo. Una de estas ideas fue la rueda de Jefferson, concebida por el tercer presidente de los Estados Unidos, Thomas Jefferson. Este dispositivo constaba de 36 discos de madera numerados que formaban un cilindro y tenían la capacidad de girar libremente. En la parte exterior de cada disco se disponían las letras del alfabeto en órdenes diferentes, asegurando así que no existieran dos discos idénticos. La clave de cifrado residía en el orden en el que se ensamblaban los discos, un acuerdo que debían establecer tanto el emisor como el receptor para cifrar y descifrar los mensajes. Para llevar a cabo el cifrado, bastaba con dividir el texto claro en bloques de 36 letras, girar los discos hasta formar el mensaje a lo largo de una de las líneas y seleccionar cualquier otra fila de letras. Aunque la rueda de Jefferson nunca llegó a ser empleada, sentó las bases para el diseño de las máquinas de escribir que integraban el cifrado, las máquinas de rotores o cifradores de rotación.

En 1915, dos oficiales holandeses, Theo A. van Hengel y R.P.C. Spengler, idearon una máquina de rotores para poder cifrar mensajes. Construyeron un prototipo pero la armada holandesa rechazó la idea al no considerarlo algo necesario para sus comunicaciones y evitó la patente del dispositivo. Estos hechos no serían descubiertos hasta el año 2003 provocando que no se les otorgara el reconocimiento de la creación de la primera máquina de rotores hasta esta fecha.

Hasta el año 2003, se consideró a Edward Hugh Hebern, inventor estadounidense nacido en 1869, el inventor de la primera máquina de cifrado de rotores. Desde 1912 había estado trabajando en varios dispositivos de cifrado hasta que en 1917 creó una máquina con un único rotor que podía ser insertado cambiando su orientación para cifrar y descifrar. En 1921 patentó su diseño añadiendo nuevos rotores y fundando Hebern Electric Code Company pero la venta de únicamente una docena de dispositivos provocó su bancarrota y su entrada en prisión acusado de fraude por sus inversores. La máquina de Hebern presentaba un rotor que giraba mientras los demás permanecían fijos, lo que la hacía vulnerable al criptoanálisis al limitar su capacidad de cifrado a solo 26 caracteres. A raíz de este invento, el criptoanalista estadounidense William Friedman mejoró el diseño original mediante la introducción de una rotación irregular en los rotores, dando lugar a la creación de la máquina SIGABA o ECM Mark II, que ofrecía una mayor seguridad en el cifrado.

Capítulo 4. Criptografía en la Segunda Guerra Mundial



Figura 4.1: Máquina de Hebern. Fuente: Computer History Museum [14]

El ingeniero alemán Arthur Scherbius se propuso estudiar la sustitución de los sistemas criptográficos manuales de la Primera Guerra Mundial por una máquina capaz de mejorar las posibilidades de cifrado y, al mismo tiempo, simplificar la labor del emisor. Finalmente, desarrolló una máquina criptográfica que fue considerada una versión mecánica del disco Alberti, a la que denominó Enigma y para la cual solicitó la primera patente en 1918. Su principal objetivo era mantener en secreto los principales intercambios de información entre empresarios, pero nadie podía sospechar que acabaría convirtiéndose en el sistema de codificación militar más temido de la historia. Más adelante se tratará su funcionamiento e historia con más detalle.



Figura 4.2: Máquina Enigma. Fuente: National Geographic [15]

4.3. La máquina Enigma

En 1919, el inventor holandés Hugo Alexander Koch patentó su propia versión de la máquina de rotores. Se cree con alta probabilidad que obtuvo la idea para su creación a través de su hermanastro, quien era uno de los abogados que inició el proceso de patente de Hengler y Spengler. Koch no comercializó ninguno de sus ejemplares, lo que eventualmente lo llevó a vender parte de los derechos de su máquina a Scherbius en 1927. Durante el desarrollo de la máquina Enigma, Koch colaboró estrechamente con Scherbius, lo que ha llevado a la hipótesis de que el alemán realizó dicha compra para proteger su propia invención.

Otra máquina de cifrado de rotores fue creada por el inventor sueco Arvid Gerhard Damm, patentando su diseño unos días después de Koch y creando la empresa Aktiebolaget Cryptograph. Su diseño destacaba por un rotor doble con cadencia irregular, aunque su elevada tasa de errores provocó que su comercialización inicial no fuera exitosa. Damm contaba con dos inversores mayoritarios, Karl Wilhelm Hagelin y Emanuel Nobel. El hijo de Karl, Boris, se graduó en ingeniería mecánica y se unió a la empresa, asumiendo su liderazgo tras la muerte de Damm. Boris Hagelin desarrolló máquinas exitosas como la B-211, una máquina capaz de imprimir, o la C-35, un diseño portátil. Pero, sin duda, la C-38 sería su mayor creación llegando a ser adquirida por el gobierno estadounidense que la rebautizaría como la M-209. Durante la Segunda Guerra Mundial se vendieron más de 140000 unidades consolidando a Boris Hagelin como el único triunfador en el ámbito empresarial con las máquinas de cifrado.



Figura 4.3: Máquina de cifrado M-209. Fuente: JFBouch

4.3. La máquina Enigma

El 23 de febrero de 1918 Arthur Scherbius y Richard Ritter solicitaron la patente de una máquina de cifrado que empleaba discos móviles que variaban de posición durante el cifrado. La idea de la máquina surgió con el propósito de reemplazar el cifrado convencional de forma manual por un cifrado mecánico más seguro y menos laborioso. Sin embargo, su desarrollo no llegó a tiempo para su utilización en la Primera Guerra Mundial.

Capítulo 4. Criptografía en la Segunda Guerra Mundial

La máquina Enigma utilizaba el concepto de los discos cifradores sobre el que había trabajado Alberti en el siglo XV, pero los nuevos avances tecnológicos permitían facilitar su uso e incorporar variaciones que aumentaban la seguridad. Se trataba de un sistema parecido a una máquina de escribir que incorporaba una batería y no utilizaba papel. Su modelo constaba de un teclado con 26 caracteres, correspondientes a las 26 letras del alfabeto, un clavijero interno con 6 pares de conexiones cableadas intercambiables, un panel luminoso con los 26 caracteres, 1 cilindro de entrada, 3 discos o rotores con 26 ranuras en su perímetro (dentro de cada rotor se conectan las entradas y salidas de forma irregular) y un reflector para devolver el impulso eléctrico a los rotores una vez que la señal ya había sido cifrada.

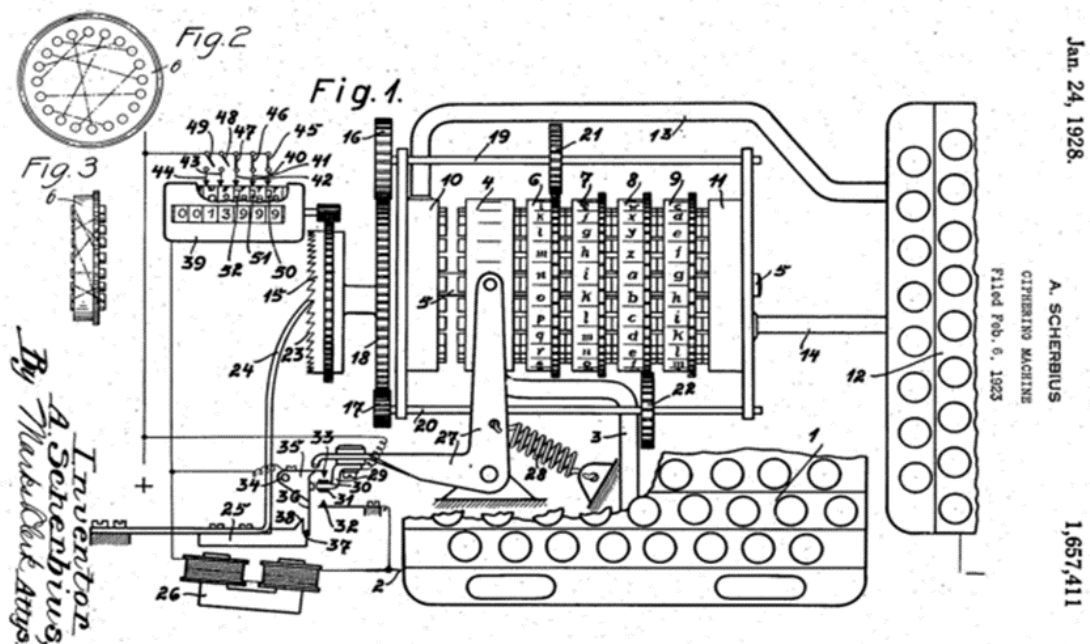


Figura 4.4: Patente estadounidense de Enigma por Arthur Scherbius, 1928. Fuente: Wikipedia [16]



Figura 4.5: Ejemplo de rotor. Fuente: La cripta del hacker [17]

4.3. La máquina Enigma

Su funcionamiento era sencillo: el usuario pulsaba una letra del teclado y se enviaba un impulso eléctrico al clavijero, en el que las conexiones redirigían el impulso hasta los rotores. El impulso se propagaba de derecha a izquierda a través de los cables de cada rotor y, tras ser reflejado, regresaba de izquierda a derecha. Con cada pulsación de tecla por parte del usuario, el rotor derecho avanzaba una posición en la máquina, lo que contribuía a evitar la repetición de cifrados. Después de que el primer rotor completaba una vuelta, pasaba a rotar el rotor a su izquierda, y así sucesivamente. Este mecanismo aseguraba que los ciclos antes de volver a la posición inicial fueran considerablemente más largos. Finalmente, el impulso llegaba al clavijero y se redirigía al panel luminoso iluminando la letra correspondiente al cifrado.

Un ejemplo más visual del funcionamiento se refleja en la figura 4.6:

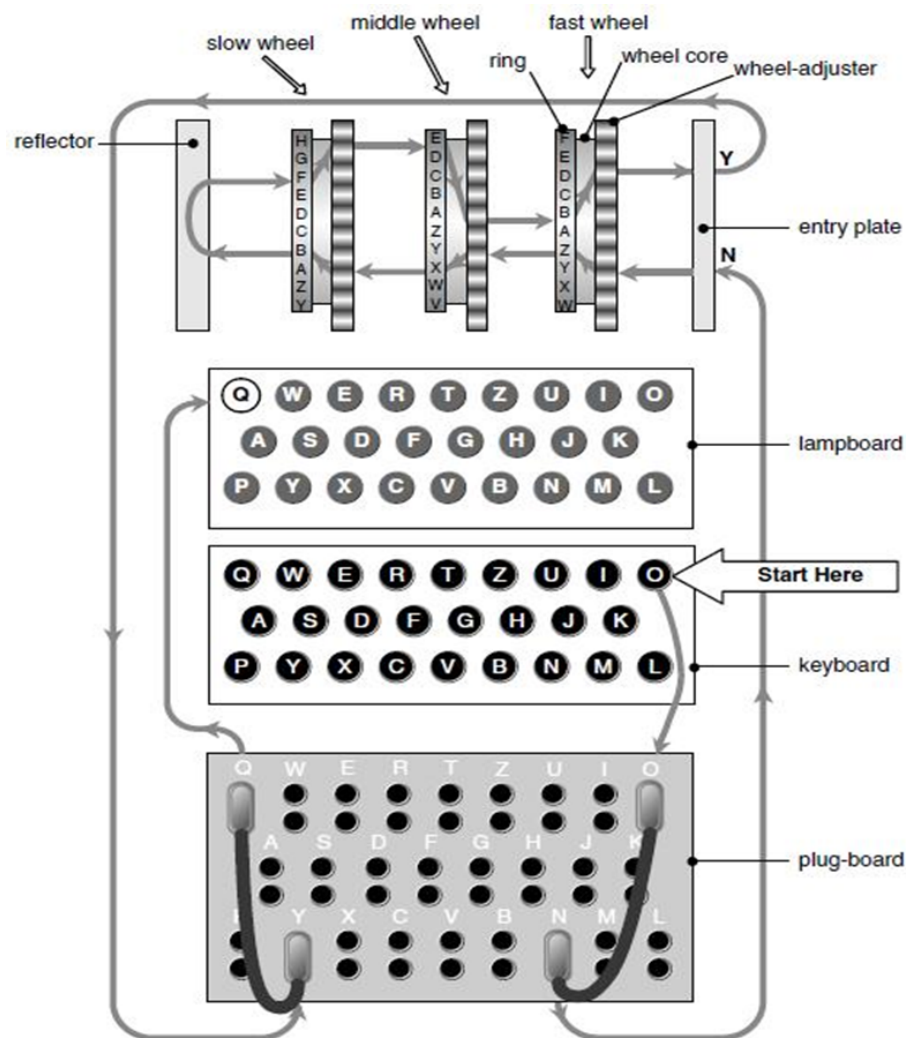


Figura 4.6: Ejemplo de cifrado en Enigma. Fuente: Francis Naukas [18]

El proceso comienza con la pulsación de la letra O en el teclado. El impulso eléc-

Capítulo 4. Criptografía en la Segunda Guerra Mundial

trico se transfiere al clavijero, donde se establece una conexión entre la clavija de la letra O y la clavija de la letra N. Luego, el impulso se dirige desde la clavija N hasta el cilindro de entrada y los rotores, donde se desplaza siguiendo las conexiones internas de cada uno. Posteriormente, el impulso alcanza el reflector, desde donde es reflejado nuevamente a través de los rotores y el cilindro de entrada hasta que se obtiene la letra resultante, en este caso, la Y. El impulso regresa al clavijero, y dado que hay una conexión entre la clavija Q y la clavija Y, finalmente se ilumina la letra Q en el panel.

En la máquina Enigma, la clave estaba compuesta por el orden de los rotores que se insertaban en la máquina, sus conexiones internas, la posición inicial de cada rotor y las conexiones del clavijero. Para intercambiar mensajes de forma segura, tanto emisor como receptor debían tener la misma configuración. A diferencia del teletipo de Vernam, la máquina Enigma no contaba con un método integrado de transmisión. Después de codificar el mensaje, este se traducía a código morse y se transmitía a través de la línea de comunicación.

Tanto la Armada Alemana como el Ejército, en 1926 y 1928 respectivamente, encargaron máquinas Enigma a la empresa de Ritter y Scherbius. El coste de estas máquinas representó un gasto significativo para el Imperio Alemán en esa época, al juntarse con sus obligaciones impuestas en el Tratado de Versalles. Los alemanes solicitaron diversas modificaciones para incrementar la seguridad y la complejidad de los modelos empleados exclusivamente para propósitos militares. Algunas de estas modificaciones se enfocaron en el clavijero, característica ausente en la versión de la máquina Enigma destinada al público general, o en la exclusividad de las conexiones internas de cada rotor. Esta preparación criptográfica inicial junto con la reconstrucción del ejército en la sombra le otorgó una ventaja significativa en el inicio del conflicto en 1939.

Durante el conflicto, la preocupación alemana por la seguridad fue máxima ante los fallos cometidos en la Primera Guerra Mundial. Para ello, cada mes los operadores de Enigma recibían un libro del alto mando con las claves diarias a utilizar en dicho mes. Un ejemplo de clave diaria es el siguiente:

- Walzenlage: III – I – II
- Ringstellung: A – D – J
- Stecker: A/F, C/I, J/P, L/H, V/X, R/Z

El operador que recibiera la anterior configuración debía colocar el tercer rotor en el primer hueco, el primer rotor en el segundo hueco y el segundo rotor en el tercer hueco (Walzenlage). En segundo lugar, girar cada uno de los rotores hasta que la letra visible fuera la A para el rotor en el primer hueco, la D para el rotor en el segundo hueco y la J para el rotor en el tercer hueco (Ringstellung). Finalmente, el operador debía conectar los cables uniendo los pares de letras descritos en el clavijero (Stecker).

Los alemanes adelantaron que en período de conflicto el tráfico de comunicaciones podía ser inmenso y con una misma configuración se generarían infinidad de mensajes facilitando el trabajo de los criptoanalistas. Ante esta situación,

4.4. El criptoanálisis polaco de la máquina Enigma

decidieron incluir una serie de normas estrictas entre las que se incluyen:

1. En el clavijero no se debe conectar una letra con su inmediatamente posterior o anterior.
2. Un rotor no podía permanecer en el mismo hueco más de un día.
3. Se debía generar una “clave de mensaje”, es decir, cada mensaje debía tener su propia clave. Para ello, el operador debía preparar la máquina con la configuración diaria y escribir tres letras elegidas al azar, por ejemplo ABC. Se obtendría su respectivo cifrado en el panel luminoso, se supone LSZ, y se volvían a teclear las tres letras iniciales, ABC, para evitar errores humanos o por interferencias de la transmisión. Una vez introducidas de nuevo, se obtienen unas nuevas tres letras en el panel luminoso, se suponen GPR. Tras escribir las letras LSZGPR se procedía a rotar los discos hasta que fueran visibles las letras ABC y se empezaba a cifrar el mensaje. El receptor del mensaje, con la misma configuración inicial que el emisor, procedía a escribir las letras LSZGPR recibiendo en el panel luminoso ABCABC, y ajustaba la posición inicial de los discos para seguir descifrando el mensaje. Esto se conocía como Grundstellung.

Estas normas impuestas por los alemanes consiguieron el efecto contrario, ya que lejos de reforzar la seguridad otorgaron pistas y un punto inicial a los aliados para comenzar a atacar el cifrado. También es posible observar que el sistema empleado por los alemanes con la máquina Enigma incumplía uno de los principios más importantes de Kerckhoffs: La clave debe ser memorizable y no se debe recurrir a notas escritas.

4.4. El criptoanálisis polaco de la máquina Enigma

Tras la Primera Guerra Mundial y la firma del Tratado de Versalles, los aliados vigilaban las comunicaciones alemanas con el objetivo de evitar un nuevo conflicto bélico. A partir de 1926 se comenzaron a interceptar mensajes cifrados con un método desconocido para ellos hasta ese momento, la versión de la máquina Enigma construida exclusivamente para las operaciones militares alemanas. Este suceso provocó que la mayoría de países entrase en alerta y se empezaran a aplicar técnicas de criptoanálisis para obtener el texto claro sin éxito.

El estado soberano de Polonia, recién formado en 1918, se encontraba ubicado entre la Unión Soviética, cuyo objetivo era la expansión del comunismo, y Alemania, que ansiaba por recuperar los territorios cedidos. Ante esta situación de peligro, la alianza de la Oficina de Radio-Inteligencia y la Oficina Criptográfica Polaca desembocó en la creación de la Oficina de Cifras (Biuro Szyfrów). El máximo responsable de la Oficina era Gwido Langer, teniente coronel de la 2ª sección del ejército polaco, y el encargado de la sección dedicada a los mensajes alemanes era Maksymilian Ciężki, capitán que había participado en la Primera Guerra Mundial en el bando alemán. A pesar de poseer conocimientos sobre la máquina Enigma comercializada al público, Ciężki no fue capaz de obtener el texto claro debido a su desconocimiento de los cambios introducidos para las

Capítulo 4. Criptografía en la Segunda Guerra Mundial

operaciones militares alemanas.

Desde los comienzos de la criptografía había existido una estrecha vinculación con las matemáticas, la cual había ido intensificándose a medida que aumentaba la complejidad de los sistemas. El director de la Universidad de Poznań suministró una lista de 20 estudiantes de los últimos cursos de matemáticas para formar parte un curso de criptología organizado por el ejército con el propósito de descifrar los mensajes alemanes. En este curso se llevaron a cabo diversas pruebas que solo tres estudiantes fueron capaces de superar: Marian Rejewski, Jerzy Ròżycki y Henrik Zygalski. Tanto Ròżycki como Zygalski fueron invitados a colaborar con la Oficina de Cifras tras la finalización del curso. Por su parte, Rejewski obtuvo una beca para estudiar en la Universidad de Gotinga, en la que habían impartido clases figuras matemáticas como Gauss, Dirichlet o Riemann. Por tanto, no fue hasta otoño de 1930 que comenzó a ayudar a la Oficina. Este grupo de jóvenes matemáticos estableció su base en los sótanos de la comandancia militar polaca, cercanos al Instituto de Matemáticas de Poznań ya que lo compaginaban con sus estudios universitarios. Aunque lograron descifrar numerosos códigos, no tuvieron éxito inicialmente con el cifrado de la máquina Enigma. En 1932, dado que el uso de Enigma se había generalizado en todo el ejército alemán, las autoridades militares polacas consideraron oportuno profesionalizar a los tres estudiantes, primero a Rejewski y años más tarde a Zygalski y Ròżycki, y ofrecerles empleo en el Biuro Szyfrów de Varsovia. Así nacía el grupo de criptoanalistas BS4.



Figura 4.7: Henryk Zygalski, Jerzy Ròżycki y Marian Rejewski, 1932. Fuente: Museumsstiftung Post und Telekommunikation

4.4. El criptoanálisis polaco de la máquina Enigma

En 1932, Rejewski había empezado a trabajar a partir de la clave del mensaje, cifrada dos veces al principio de cada comunicación, pero el desconocimiento del funcionamiento interno de la máquina obstaculizaba su avance. Esta situación cambió con la intervención de Hans Thilo Schmidt, exfuncionario del ejército alemán expulsado tras los recortes de la Primera Guerra Mundial y que había comenzado a trabajar en la oficina responsable de comunicaciones gracias a su hermano Rudolph. Su situación de precariedad y el resentimiento por haber sido expulsado le llevó a acceder a intercambiar información valiosa con la embajada francesa a cambio de una compensación económica. Durante múltiples encuentros entre la embajada francesa y Schmidt, con nombre en clave Asché, este aportó los libros del alto mando con todas las claves completas de 38 meses y fotos de la máquina Enigma de la que hacía uso el ejército alemán, sin ofrecer detalles sobre el cableado interno. Los criptoanalistas franceses no lograron aprovechar esta información, por lo que decidirían compartirla con los polacos a finales de 1932.

4.4.1. Estudio de las permutaciones

Previamente, los criptoanalistas polacos contaban con una máquina Enigma comercial con la que habían estado trabajando, así como cientos de mensajes interceptados para su análisis. La incorporación de los libros de códigos y las imágenes de la máquina les permitió teóricamente reconstruir el funcionamiento de la máquina Enigma militar, basándose en las permutaciones de la teoría de grupos. Una permutación también puede ser definida como un cambio de orden en un conjunto, en el caso criptográfico con el conjunto de letras a,b,c,d,e y f un ejemplo es el siguiente:

- La letra a pasa a convertirse en la b.
- La letra b pasa a convertirse en la c.
- La letra c pasa a convertirse en la a.
- La letra d pasa a convertirse en la e.
- La letra e pasa a convertirse en la f.
- La letra f pasa a convertirse en la d.

La permutación, a la que se llamará P , transforma el conjunto "abcdef" en el conjunto "bcaefd". Se tienen dos ciclos disjuntos por lo que la permutación P puede expresarse de la forma:

$$P = (abc)(def)$$

A continuación, se explican algunos aspectos clave de la teoría de permutaciones que se utilizarán para llevar a cabo la explicación.

Si X es un conjunto no vacío, se dice que una permutación de X es una aplicación biyectiva $\alpha : X \rightarrow X$. Se denota el conjunto de todas las permutaciones de X por S_X . Sea θ una de las permutaciones de S_x , es posible representarla con una matriz de correspondencia de la siguiente forma:

Capítulo 4. Criptografía en la Segunda Guerra Mundial

$$\theta = \begin{pmatrix} x_1 & x_2 & \dots & x_n \\ x_{i1} & x_{i2} & \dots & x_{in} \end{pmatrix}$$

donde $\theta x_1 = x_{i1}$, es decir, el elemento x_1 se convierte en x_{i1} y así sucesivamente. La correspondencia que establece θ es biyectiva al existir una aplicación inversa θ^{-1} que verifica que $\theta \circ \theta^{-1} = I$, siendo I la aplicación identidad. Un ejemplo es el siguiente:

$$\theta = \begin{pmatrix} a & b & c & d & e & f \\ b & c & a & e & f & d \end{pmatrix}; \theta^{-1} = \begin{pmatrix} b & c & a & e & f & d \\ a & b & c & d & e & f \end{pmatrix} = \begin{pmatrix} a & b & c & d & e & f \\ c & a & b & f & d & e \end{pmatrix}$$

$$\theta \circ \theta^{-1} = \begin{pmatrix} a & b & c & d & e & f \\ a & b & c & d & e & f \end{pmatrix} = I$$

Una permutación del conjunto S_x puede ser representada sin ambigüedad por una permutación del conjunto $\{1, 2, \dots, n\}$, denotándose en ese caso el conjunto S_x por S_n y denominándose Grupo simétrico de grado n .

Sean γ y θ dos permutaciones de S_n , el producto $\gamma\theta$ se interpreta como una composición de n aplicaciones, $\gamma\theta(m) = \gamma(\theta(m))$, $\forall m \in \{1, 2, \dots, n\}$. Un ejemplo que demuestra que dicha operación no es conmutativa es el siguiente:

$$\gamma = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 \\ 2 & 3 & 4 & 5 & 6 & 1 \end{pmatrix}, \theta = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 \\ 2 & 3 & 1 & 5 & 6 & 4 \end{pmatrix}$$

$$\gamma\theta = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 \\ 2 & 3 & 4 & 5 & 6 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 \\ 2 & 3 & 1 & 5 & 6 & 4 \end{pmatrix} = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 \\ 3 & 4 & 2 & 6 & 1 & 5 \end{pmatrix}$$

$$\theta\gamma = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 \\ 2 & 3 & 1 & 5 & 6 & 4 \end{pmatrix} \cdot \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 \\ 2 & 3 & 4 & 5 & 6 & 1 \end{pmatrix} = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 \\ 3 & 1 & 5 & 6 & 4 & 2 \end{pmatrix}$$

Se denomina soporte de la permutación, notado por A_θ , al conjunto de elementos $\{1, 2, \dots, n\}$ que son movidos por una permutación θ . Sean dos permutaciones γ y θ , se dice que son disjuntas si $A_\gamma \cap A_\theta = \emptyset$. Un ejemplo es el siguiente:

$$\gamma = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 \\ 1 & 2 & 3 & 5 & 6 & 4 \end{pmatrix}; \theta = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 \\ 2 & 3 & 1 & 4 & 5 & 6 \end{pmatrix} \Rightarrow$$

$$\Rightarrow A_\gamma = \{4, 5, 6\}, A_\theta = \{1, 2, 3\} \Rightarrow A_\gamma \cap A_\theta = \emptyset$$

Teorema 1. Si γ es una permutación de S_n entonces la imagen por γ de todos los elementos de A_γ está en A_γ .

Demostración. Sea $x \in A_\gamma$, entonces $\gamma(x) = y$ con $x \neq y$. Como γ es inyectiva y $x \neq y \Rightarrow \gamma(y) \neq \gamma(x) = y \Rightarrow \gamma(y) = z \neq y \Rightarrow y \in A_\gamma$. $\square$

4.4. El criptoanálisis polaco de la máquina Enigma

Teorema 2. Si γ y θ son permutaciones disjuntas en S_n , entonces $\gamma\theta = \theta\gamma$, es decir, conmutan.

Demostración. Dado que γ y θ son permutaciones disjuntas, se verifica que $A_\gamma \cap A_\theta = \emptyset$. Sea $A_\gamma = \{a_1, \dots, a_s\}$, $A_\theta = \{b_1, \dots, b_s\}$ y $\{1, \dots, n\} = A_\gamma \cup A_\theta \cup \{c_1, \dots, c_t\} \Rightarrow$

$$\begin{cases} \forall x \in A_\gamma & \gamma(x) \in A_\gamma \Rightarrow \theta(\gamma(x)) = \gamma(x) = \gamma(\theta(x)) \\ \forall y \in A_\theta & \theta(y) \in A_\theta \Rightarrow \gamma(\theta(y)) = \theta(y) = \theta(\gamma(y)) \\ \forall z \in \{c_1, \dots, c_t\} & \gamma(\theta(z)) = \gamma(z) = z = \theta(z) = \theta(\gamma(z)) \end{cases}$$

Por tanto $\forall x \in \{1, \dots, n\}$ se verifica que $\gamma(\theta(x)) = \theta(\gamma(x))$

Esto demuestra que $\gamma\theta = \theta\gamma$ si γ y θ son permutaciones disjuntas en S_n . $\square$

Se denomina ciclo a una permutación $\theta \in S_n$ si existen elementos $s_1, s_2, \dots, s_k$ en el conjunto $\{1, 2, \dots, n\}$ tales que:

1. Se tienen las relaciones $\theta(s_1) = s_2, \theta(s_2) = s_3, \dots, \theta(s_{k-1}) = s_k$ y $\theta(s_k) = s_1$.
2. La permutación θ deja fijos todos los elementos distintos de s_i pertenecientes a $\{1, 2, \dots, n\}$.

Teorema 3. Toda permutación es un ciclo, o puede descomponerse como un producto de ciclos disjuntos. Esta descomposición es única salvo por el orden de los factores.

Demostración. Sea $\gamma \in S_n$, $G = \langle \gamma \rangle$ el subgrupo de S_n generado por γ y $X = \{1, \dots, n\}$ entonces G actúa sobre X . Se conoce como órbita de una permutación γ sobre $a \in X = \{1, \dots, n\}$ a:

$$G_\gamma a = \{a, \gamma(a), \gamma^2(a), \dots\} = \{\gamma^z(a) \mid z \in \mathbb{N}\}$$

y se sabe que todo elemento de X pertenece a una órbita ya que $\forall b \in X$ es $b \in G_\gamma b$ y que el conjunto de todas las órbitas de una permutación γ establece una partición en X en el sentido de que las órbitas de γ correspondientes a dos elementos distintos o bien coinciden, o son disjuntas y además la unión de todas las órbitas es todo X (se omite la demostración de estos resultados pero pueden ser encontradas en Abstract Algebra: An Integrated Approach, Joseph H. Silverman [19]). De esta forma, X se puede escribir como una unión disjunta de órbitas de la forma $X = G_\gamma a_1 \cup G_\gamma a_2 \cup \dots \cup G_\gamma a_r$ con $a_i \in X, i \in \{1, \dots, r\}$ donde r es el número de órbitas tales que $G_\gamma a_i \cap G_\gamma a_j = \emptyset$ para $i \neq j$ con $i, j \in \{1, \dots, r\}$.

Llamando $k_i = \#G_\gamma a_i$ al cardinal de la órbita $G_\gamma a_i$, la órbita será exactamente:

$$G_\gamma a_i = \{a_i, \gamma(a_i), \gamma^2(a_i), \dots, \gamma^{k_i-1}(a_i)\}$$

Por tanto, el ciclo σ_i definido por

$$\sigma_i = (a_i, \gamma(a_i), \gamma^2(a_i), \dots, \gamma^{k_i-1}(a_i)) \text{ es el } \gamma\text{-ciclo que contiene } a_i$$

Capítulo 4. Criptografía en la Segunda Guerra Mundial

Para $i \neq j$, los ciclos σ_i y σ_j asociados a las órbitas Ga_i y Ga_j son disjuntos (por serlo sus órbitas asociadas), esto provoca que si se multiplica $\sigma_1, \dots, \sigma_r$, se obtiene una permutación que es igual a γ ,

$$\gamma = \sigma_1 \sigma_2 \dots \sigma_r.$$

Esto prueba que toda permutación es igual a un producto de ciclos disjuntos. Para demostrar que es única salvo por el orden basta con apreciar que dado que los ciclos son disjuntos, todo $a \in \{1, 2, \dots, n\}$ aparece exactamente en uno de los ciclos y su sucesor debe ser $\gamma(a)$ en dicho ciclo. De este modo, la descomposición es única salvo por el orden. $\square$

Sea $\sigma = \sigma_1 \sigma_2 \dots \sigma_t$ un producto de ciclos disjuntos con longitudes $m_1, m_2, \dots, m_t$ respectivamente. El orden de la permutación σ es el $mcm(m_1, m_2, \dots, m_t)$.

Sea $\sigma = (i_1, i_2, \dots, i_m)$ un ciclo de longitud m , entonces $\sigma^{-1} = (i_m, \dots, i_2, i_1)$ también es un ciclo y tiene longitud m .

Sea $\sigma = \sigma_1 \sigma_2 \dots \sigma_t$ un producto de ciclos disjuntos con longitudes $m_1, m_2, \dots, m_t$. Entonces $\sigma^{-1} = \sigma_1^{-1} \sigma_2^{-1} \dots \sigma_t^{-1}$.

Un ciclo de longitud 2 se denomina transposición. Un ejemplo es $\sigma = (ab)$ en el que a se convierte en b siendo $a, b \in \{1, 2, \dots, n\}$.

Teorema 4. *Toda permutación se puede descomponer como un producto de transposiciones. La descomposición no es única.*

Demostración. Se sabe del Teorema 3 que cualquier permutación γ puede ser escrita como un producto de ciclos disjuntos. Será suficiente con probar que cada ciclo $\sigma = (a_1, a_2, \dots, a_k)$ puede ser representado como un producto de transposiciones. Demostremos que la permutación

$$v = (a_k, a_1)(a_{k-1}, a_1) \dots (a_3, a_1)(a_2, a_1)$$

es igual al ciclo σ . Para los valores de $v(a_i)$ se considerarán tres casos:

1. $v(a_1)$: Al aplicar v sobre a_1 , la primera transposición (a_2, a_1) transforma a_1 en a_2 y ninguna otra transposición de v tiene efecto sobre a_2 . Por tanto, $v(a_1) = a_2$.
2. $v(a_i)$ con $2 \leq i \leq k-1$: La primera transposición de v que actúa sobre a_i es (a_i, a_1) , que transforma a_i en a_1 . A continuación, la transposición (a_{i+1}, a_1) transforma a_1 en a_{i+1} y el resto de transposiciones no tienen efecto sobre a_{i+1} . Por tanto, $v(a_i) = a_{i+1}$.
3. $v(a_k)$: La única transposición de v que actúa sobre a_k es la última, (a_k, a_1) , transformando a_k en a_1 . Por tanto, $v(a_k) = a_1$.

Se ha demostrado que $v(a_i) = a_{i+1}$ para $1 \leq i \leq k-1$ y que $v(a_k) = a_1$. De este modo, v es exactamente el ciclo σ finalizando la demostración [19]. $\square$

Un ejemplo del resultado proporcionado por 4 es el siguiente: La permutación cíclica $\sigma = (abc)$ también se puede expresar como $\sigma = (ab)(bc)(ca)$.

4.4. El criptoanálisis polaco de la máquina Enigma

Teorema 5. Sea la permutación $\theta \in S_n$. Entonces θ no puede ser descompuesta como un producto de un número par e impar de transposiciones simultáneamente.

Se omite la demostración de este resultado, pero puede ser consultado en Abstract Algebra: An Integrated Approach [19].

Dos permutaciones γ y θ en S_n , se dirá que son **conjugadas** si existe otra permutación $\psi \in S_n$ tal que

$$\theta = \psi\gamma\psi^{-1}$$

Se conoce como **tipo o estructura formal** de la permutación al producto formal $1^{m_1}2^{m_2}\dots k^{m_k}$, donde para cada $i \in \{1, \dots, k\}$, m_i son los ciclos disjuntos de longitud i en los que se descompone la permutación.

Teorema 6. Dos permutaciones son conjugadas si y sólo si son del mismo tipo o poseen la misma estructura formal.

Demostración. Sean α y β dos permutaciones expresadas como producto de ciclos disjuntos y suponiendo que $\alpha(a_1) = a_2$, entonces se verifica que:

$$\beta\alpha\beta^{-1}(\beta(a_1)) = \beta\alpha(\beta^{-1}(\beta(a_1))) = \beta(\alpha(a_1)) = \beta(a_2)$$

$\Rightarrow$) Al ser β una biyección, cada ciclo $(a_1, a_2, \dots)$ en la representación por ciclos disjuntos de α corresponde a un ciclo $(\beta(a_1), \beta(a_2), \dots)$ en la representación por ciclos disjuntos de $\beta\alpha\beta^{-1}$. De este modo α y $\beta\alpha\beta^{-1}$ tendrán la misma estructura en su representación por ciclos disjuntos, es decir, la misma estructura formal.
 $\Leftarrow$) Si α y una permutación γ poseen la misma estructura formal, entonces para cada ciclo $(a_1, a_2, \dots)$ en la representación por ciclos disjuntos de α debe existir una correspondencia $(g_1, g_2, \dots)$ en la representación por ciclos disjuntos de γ . Si se define β con $\beta(a_1) = g_1, \beta(a_2) = g_2$ y así para todos los ciclos disjuntos en la representación por ciclos disjuntos de α , entonces $\beta\alpha\beta^{-1} = \gamma$. [20] $\square$

Un ejemplo del resultado proporcionado por el Teorema 6 se muestra en estas dos permutaciones expresadas por su notación cíclica:

$$\gamma = (af)(bc)(deg); \theta = (ad)(be)(cfg)$$

Ambas poseen una estructura cíclica idéntica. Por tanto, se podrá definir una permutación ψ situando una permutación encima de la otra:

$$\psi = \begin{pmatrix} a & b & c & d & e & f & g \\ a & b & e & c & f & d & g \end{pmatrix} \Rightarrow$$

$$\begin{aligned} \Rightarrow \psi\gamma\psi^{-1} &= \begin{pmatrix} a & b & c & d & e & f & g \\ a & b & e & c & f & d & g \end{pmatrix} \cdot \begin{pmatrix} a & b & c & d & e & f & g \\ f & c & b & e & g & a & d \end{pmatrix} \cdot \begin{pmatrix} a & b & c & d & e & f & g \\ a & b & d & f & c & e & g \end{pmatrix} = \\ &= \begin{pmatrix} a & b & c & d & e & f & g \\ a & b & e & c & f & d & g \end{pmatrix} \cdot \begin{pmatrix} a & b & c & d & e & f & g \\ f & c & e & a & b & g & d \end{pmatrix} = \begin{pmatrix} a & b & c & d & e & f & g \\ d & e & f & a & b & g & c \end{pmatrix} = \theta \end{aligned}$$

Capítulo 4. Criptografía en la Segunda Guerra Mundial

La fuente de estas nociones de permutación es Historia de Matemáticas: Criptología Nazi. Los Códigos Secretos de Hitler, José Manuel Sánchez Muñoz [4].

Rejewski logró establecer una correlación entre el funcionamiento de la máquina Enigma y el fundamento teórico de las permutaciones. En cada instancia que el operador pulsaba una tecla de la máquina, el impulso eléctrico se distribuía por el clavijero, la entrada del cilindro, los tres rotores, el reflector, los tres rotores nuevamente, la entrada del cilindro y, finalmente, el clavijero hasta encenderse la luz del panel. Cada uno de estos componentes llevaba a cabo una permutación, es decir, una transformación de letras. Haciendo uso de la siguiente notación:

- S : permutación causada por el clavijero.
- H : permutación causada por el cilindro de entrada.
- N : permutación causada por el rotor derecho.
- M : permutación causada por el rotor central.
- L : permutación causada por el rotor izquierdo.
- R : permutación causada por el reflector.

es posible definir el camino del impulso eléctrico mediante un producto de permutaciones, en este caso:

1. El impulso llegaba al clavijero, representado con la permutación S .
2. Después del clavijero llegaba al cilindro de entrada, representado con la permutación H .
3. El impulso pasaba por los tres rotores, representado con el producto de permutaciones LMN .
4. Se producía el reflejo en el reflector, representado por la permutación R .
5. Volvía por los tres rotores en sentido inverso, lo que significaba que las sustituciones se realizaban de manera inversa (si antes la A se transformaba en la B, ahora la B se transforma en la A). Por tanto, se representaba con el producto de permutaciones $N^{-1}M^{-1}L^{-1}$.
6. Continuaba pasando nuevamente por el cilindro de entrada, representado por la permutación H^{-1} por ser en el camino inverso.
7. Finalmente, el impulso pasaba por el clavijero, representado por la permutación S^{-1} , y llegaba al panel. La permutación también podría representarse como S ya que en el caso del clavijero, como las sustituciones eran de la forma X se transforma en Y e Y se transforma en X (ciclos de orden 2 o transposiciones disjuntas), entonces $S = S^{-1}$.

Entonces, el producto de permutaciones que representaba el efecto de pulsar una letra era:

$$S^{-1}H^{-1}N^{-1}M^{-1}L^{-1}RLMNHS$$

4.4. El criptoanálisis polaco de la máquina Enigma

Agrupando la permutación global de Enigma, es posible expresar dicha permutación como una permutación conjugada del reflector:

$$(LMNHS)^{-1}R(LMNHS)$$

El reflector era un medio rotor, lo que provocaba que los 26 contactos correspondientes a cada una de las letras estuviesen conectados por parejas. Ante este funcionamiento, fue posible definir que la permutación del reflector consistía en 13 transposiciones disjuntas, y, haciendo uso del Teorema (6), la permutación global de Enigma debía poseer la misma estructura formal.

Una vez establecido el producto de permutaciones global de Enigma y su estructura formal, Rejewski tuvo que tener en cuenta que cada vez que el usuario pulsaba una tecla el rotor derecho giraba una posición antes de completar el circuito eléctrico. Esto llevó a Rejewski a contemplar una nueva permutación en la que cada letra se transformaba en su sucesora, la cual denominó P :

$$P = (\text{abcdefghijklmnopqrstuvwxyz})$$

Cada vez que el rotor giraba, se aplicaba la permutación P , seguida de la permutación N correspondiente al cableado interno del rotor, y después la inversa de la permutación P . El resultado de esta secuencia de operaciones era la composición $P^{-1}NP$. En el camino de vuelta del impulso la permutación será la inversa, es decir, $P^{-1}N^{-1}P$. Cuando el operador pulsaba una nueva letra, el rotor derecho volvía a girar produciéndose la permutación $P^{-2}NP^2$ en la ida y $P^{-2}N^{-1}P^2$ en la vuelta.

Ante estas premisas, Rejewski focalizó su investigación en las seis primeras letras, correspondientes a la clave del mensaje. La probabilidad de que ocurriera una rotación en el disco central era de $6/26$, mientras que en el disco izquierdo era considerablemente menor, por lo que optaron por no tomar en cuenta estas rotaciones en su punto de partida. Se definió el siguiente sistema de ecuaciones a partir de las seis primeras interacciones del operador, donde A equivale a la permutación tras la primera pulsación, B a la segunda, y así sucesivamente:

$$A = S^{-1}H^{-1}P^{-1}N^{-1}PM^{-1}L^{-1}RLMP^{-1}N^{-1}PHS = (LMP^{-1}NPHS)^{-1}R(LMP^{-1}NPHS)$$

$$B = S^{-1}H^{-1}P^{-2}N^{-1}P^2M^{-1}L^{-1}RLMP^{-2}N^{-1}P^2HS = (LMP^{-2}NP^2HS)^{-1}R(LMP^{-2}NP^2HS)$$

$$C = S^{-1}H^{-1}P^{-3}N^{-1}P^3M^{-1}L^{-1}RLMP^{-3}N^{-1}P^3HS = (LMP^{-3}NP^3HS)^{-1}R(LMP^{-3}NP^3HS)$$

$$D = S^{-1}H^{-1}P^{-4}N^{-1}P^4M^{-1}L^{-1}RLMP^{-4}N^{-1}P^4HS = (LMP^{-4}NP^4HS)^{-1}R(LMP^{-4}NP^4HS)$$

$$E = S^{-1}H^{-1}P^{-5}N^{-1}P^5M^{-1}L^{-1}RLMP^{-5}N^{-1}P^5HS = (LMP^{-5}NP^5HS)^{-1}R(LMP^{-5}NP^5HS)$$

$$F = S^{-1}H^{-1}P^{-6}N^{-1}P^6M^{-1}L^{-1}RLMP^{-6}N^{-1}P^6HS = (LMP^{-6}NP^6HS)^{-1}R(LMP^{-6}NP^6HS)$$

Para resolver el sistema, los polacos necesitaban conocer cada una de las permutaciones N , M , L , R , S y H . Por otro lado, tampoco conocían las permutaciones A , B , C , D , E y F , para ello sería necesario ser poseedor del texto claro y del texto cifrado pero solo se disponía del segundo. Ante esta incertidumbre, Rejewski decidió basar sus investigaciones en tres claves:

Capítulo 4. Criptografía en la Segunda Guerra Mundial

1. En primer lugar, la máquina Enigma se caracterizaba por su "reciprocidad", es decir, al pulsar una letra se recibía su cifrada y si se pulsaba su cifrada con la misma configuración se recibía la letra original. Esto permitía la demostración de que $X = X^{-1}, \forall X \in \{A, B, C, D, E, F\}$, en definitiva $\forall X \in \{A, B, C, D, E, F\}$, X es un producto disjunto de transposiciones.
2. En segundo lugar, era necesario aprovechar la debilidad que ocasionaba la repetición de claves. El operador introducía la configuración que aparecía en el libro mensual y escribía dos veces la misma palabra referente a la clave del mensaje antes de cambiar la configuración. Esto supuso un punto de partida para los criptoanalistas al permitirles demostrar que $A(x) = 1^{\text{a}}$ letra del mensaje cifrado, $D(x) = 4^{\text{a}}$ letra del mensaje cifrado, $B(y) = 2^{\text{a}}$ letra del mensaje cifrado, $E(y) = 5^{\text{a}}$ letra del mensaje cifrado, $C(z) = 3^{\text{a}}$ letra del mensaje cifrado y $F(z) = 6^{\text{a}}$ letra del mensaje cifrado, con x, y, z siendo las tres letras referentes a la clave del mensaje.
3. Aprovechando los dos resultados anteriores, la repetición de claves y la "reciprocidad", los polacos demostraron que:

$$A(1^{\text{a}} \text{ letra}) = x, D(x) = 4^{\text{a}} \text{ letra} \Rightarrow DA(1^{\text{a}} \text{ letra}) = 4^{\text{a}} \text{ letra}$$

$$B(2^{\text{a}} \text{ letra}) = y, E(y) = 5^{\text{a}} \text{ letra} \Rightarrow EB(2^{\text{a}} \text{ letra}) = 5^{\text{a}} \text{ letra}$$

$$C(3^{\text{a}} \text{ letra}) = z, F(z) = 6^{\text{a}} \text{ letra} \Rightarrow FC(3^{\text{a}} \text{ letra}) = 6^{\text{a}} \text{ letra}$$

con x, y, z siendo las tres letras de la clave en el texto claro.

A partir de estas claves se desconocían las permutaciones A, B, C, D, E y F pero se conocían parte de las permutaciones DA, EB y FC a mediante las 6 primeras letras. A lo largo del día se interceptaban centenas de mensajes con claves diferentes, lo que permitió obtener las permutaciones completas DA, EB y FC diarias a partir de las relaciones entre la primera y cuarta, segunda y quinta, y tercera y sexta letra de cada mensaje cifrado. Tomando $Q = M^{-1}L^{-1}RLM$, el sistema quedaba:

$$A = S^{-1}H^{-1}P^{-1}NPQP^{-1}NPHS$$

$$B = S^{-1}H^{-1}P^{-2}N^{-1}P^2QP^{-2}NP^2HS$$

$$C = S^{-1}H^{-1}P^{-3}N^{-1}P^3QP^{-3}NP^3HS$$

$$D = S^{-1}H^{-1}P^{-4}N^{-1}P^4QP^{-4}NP^4HS$$

$$E = S^{-1}H^{-1}P^{-5}N^{-1}P^5QP^{-5}NP^5HS$$

$$F = S^{-1}H^{-1}P^{-6}N^{-1}P^6QP^{-6}NP^6HS$$

$$DA = S^{-1}H^{-1}P^{-4}N^{-1}P^4QP^{-4}NP^3N^{-1}PQP^{-1}NPHS$$

$$EB = S^{-1}H^{-1}P^{-5}N^{-1}P^5QP^{-5}NP^3N^{-1}P^2QP^{-2}NP^2HS$$

$$FC = S^{-1}H^{-1}P^{-6}N^{-1}P^6QP^{-6}NP^3N^{-1}P^3QP^{-3}NP^3HS$$

4.4. El criptoanálisis polaco de la máquina Enigma

En este punto, Rejewski conocía las permutaciones compuestas DA, EB y FC , desconociendo las permutaciones H, S, N y $Q = M^{-1}L^{-1}RLM$. En su primer intento de descifrado se supuso que la permutación H , correspondiente al cilindro de entrada, era igual que el de la máquina Enigma comercial por lo que su siguiente paso se centró en obtener las permutaciones A, B, C, D, E y F . A partir del conocimiento de las permutaciones compuestas DA, EB y FC utilizó varios teoremas para lograr su objetivo:

Teorema 7 (Sobre el producto de transposiciones). *Si dos permutaciones del mismo tipo están factorizadas únicamente como producto de transposiciones disjuntas, entonces su producto contiene un número par de ciclos disjuntos de la misma longitud.*

El siguiente ejemplo ilustra un caso de la demostración:

Si $X = (a_1a_2)(a_3a_4)(a_5a_6)\dots(a_{2k-3}a_{2k-2})(a_{2k-1}a_{2k})$,
 e $Y = (a_2a_3)(a_4a_5)(a_6a_7)\dots(a_{2k-2}a_{2k-1})(a_{2k}a_1)$,
 entonces $YX = (a_1a_3a_5\dots a_{2k-3}a_{2k-1})(a_{2k}a_{2k-2}\dots a_6a_4a_2)$.

Teorema 8 (Recíproco al Teorema 7 sobre el producto de transposiciones). *Si una permutación de grado par se puede expresar como producto de dos ciclos disjuntos de la misma longitud, entonces la permutación puede ser considerada como un producto de dos permutaciones consistentes en transposiciones disjuntas.*

Demostración. La demostración es inmediata a partir del Teorema 7.

Dada $YX = (a_1a_3a_5\dots a_{2k-3}a_{2k-1})(a_{2k}a_{2k-2}\dots a_6a_4a_2)$,
 entonces es posible expresar $X = (a_1a_2)(a_3a_4)(a_5a_6)\dots(a_{2k-3}a_{2k-2})(a_{2k-1}a_{2k})$,
 e $Y = (a_2a_3)(a_4a_5)(a_6a_7)\dots(a_{2k-2}a_{2k-1})(a_{2k}a_1)$. $\square$

Teorema 9. *En las condiciones del Teorema 7, los elementos que forman parte de una única transposición, ya sea de la permutación X o de la permutación Y , forman parte siempre de dos ciclos distintos de la permutación compuesta YX .*

Demostración. A partir del Teorema 7 es inmediato.

Dado $X = (a_1a_2)(a_3a_4)(a_5a_6)\dots(a_{2k-3}a_{2k-2})(a_{2k-1}a_{2k})$,
 $Y = (a_2a_3)(a_4a_5)(a_6a_7)\dots(a_{2k-2}a_{2k-1})(a_{2k}a_1)$,
 y $YX = (a_1a_3a_5\dots a_{2k-3}a_{2k-1})(a_{2k}a_{2k-2}\dots a_6a_4a_2)$, es posible observar que en cualquier transposición de X ninguno de los dos elementos aparece en el mismo ciclo de YX . $\square$

Teorema 10. *Si dos elementos que se encuentran en dos ciclos diferentes de igual longitud de la permutación YX pertenecen a una misma transposición de X , entonces los cuatro elementos adyacentes a ellos (a la derecha y a la izquierda de cada uno de ellos) también forman transposiciones de X .*

Demostración. A partir de la demostración del Teorema 7 es inmediato.

Dado $X = (a_1a_2)(a_3a_4)(a_5a_6)\dots(a_{2k-3}a_{2k-2})(a_{2k-1}a_{2k})$,
 $Y = (a_2a_3)(a_4a_5)(a_6a_7)\dots(a_{2k-2}a_{2k-1})(a_{2k}a_1)$,
 y $YX = (a_1a_3a_5\dots a_{2k-3}a_{2k-1})(a_{2k}a_{2k-2}\dots a_6a_4a_2)$, es posible observar que tomando

Capítulo 4. Criptografía en la Segunda Guerra Mundial

por ejemplo los elementos a_3 y a_4 pertenecientes a dos ciclos distintos en YX que forman una transposición en X , entonces a_1 y a_2 , y a_5 y a_6 también forman transposiciones en X . $\square$

A partir de estos teoremas era posible obtener hasta una docena de soluciones para las permutaciones A, B y C , de las cuales solo una era correcta. La interceptación de mensajes cifrados con claves sencillas enviados por operarios descuidados, como tres letras iguales o letras dispuestas de forma consecutiva en el teclado, resultó un gran apoyo para determinar la solución correcta.

Rejewski no tenía certeza de si las ecuaciones de A, B, C, D, E y F permitían ser despejadas para obtener S, N y Q ya que era necesario poseer mensajes cifrados de dos días distintos en los que los rotores estuvieran en la misma posición pero se modificaran las conexiones del clavijero. Las posibilidades de posiciones, orientaciones y modelos de los rotores ascendían a un enorme número provocando que este problema fuera inviable en la práctica. Los criptoanalistas no se rindieron y fue en este instante, 9 de diciembre de 1932, cuando se empezaron a apoyar en los documentos facilitados por Asché (Hans Thilo Schmidt). A la permutación H , deducida a través de la Enigma comercial, las permutaciones DA, EB y FC , conocidas por la interceptación de mensajes, y las permutaciones A, B, C, D, E y F , deducidas por los criptoanalistas, ahora la permutación S , correspondiente al clavijero, dejaba de ser una incógnita y era despejable. Multiplicando cada una de las ecuaciones anteriores por HS por la izquierda y $S^{-1}H^{-1}$ por la derecha se obtenía:

$$\begin{aligned} HSAS^{-1}H^{-1} &= P^{-1}N^{-1}PQP^{-1}NP \\ HSBS^{-1}H^{-1} &= P^{-2}N^{-1}P^2QP^{-2}NP^2 \\ HSCS^{-1}H^{-1} &= P^{-3}N^{-1}P^3QP^{-3}NP^3 \\ HSDS^{-1}H^{-1} &= P^{-4}N^{-1}P^4QP^{-4}NP^4 \\ HSES^{-1}H^{-1} &= P^{-5}N^{-1}P^5QP^{-5}NP^5 \\ HSFS^{-1}H^{-1} &= P^{-6}N^{-1}P^6QP^{-6}NP^6 \end{aligned}$$

donde las únicas incógnitas eran las permutaciones N y Q (P correspondía a la permutación de rotación que convierte una letra en su sucesora). Se definieron las permutaciones U, V, W, X, Y y Z de la siguiente forma:

$$\begin{aligned} U &= PHAS^{-1}H^{-1}P^{-1} = N^{-1}PQP^{-1}N \\ V &= P^2HSBS^{-1}H^{-1}P^{-2} = N^{-1}P^2QP^{-2}N \\ W &= P^3HSCS^{-1}H^{-1}P^{-3} = N^{-1}P^3QP^{-3}N \\ X &= P^4HSDS^{-1}H^{-1}P^{-4} = N^{-1}P^4QP^{-4}N \\ Y &= P^5HSES^{-1}H^{-1}P^{-5} = N^{-1}P^5QP^{-5}N \\ Z &= P^6HSFS^{-1}H^{-1}P^{-6} = N^{-1}P^6QP^{-6}N \end{aligned}$$

4.4. El criptoanálisis polaco de la máquina Enigma

Y se calcularon las permutaciones compuestas:

$$\begin{aligned} VU &= N^{-1}P[PQP^{-1}Q]P^{-1}N \\ WV &= N^{-1}P^2[PQP^{-1}Q]P^{-2}N \\ XW &= N^{-1}P^3[PQP^{-1}Q]P^{-3}N \\ YX &= N^{-1}P^4[PQP^{-1}Q]P^{-4}N \\ ZY &= N^{-1}P^5[PQP^{-1}Q]P^{-5}N \end{aligned}$$

Despejando el factor $[PQP^{-1}Q]$ en una de las ecuaciones e introduciéndolo en la siguiente se obtuvo:

$$\begin{aligned} WV &= N^{-1}P^1NVUN^{-1}P^{-1}N \\ XW &= N^{-1}P^1NWVN^{-1}P^{-1}N \\ YX &= N^{-1}P^1NXWN^{-1}P^{-1}N \\ ZY &= N^{-1}PNYXN^{-1}P^{-1}N \end{aligned}$$

En este conjunto de ecuaciones, la única incógnita era la permutación $N^{-1}PN$. Cada día se obtenían decenas de soluciones para las permutaciones WV , XW , YX y ZY que mantenían una estructura común. Si las permutaciones no poseían la misma estructura, podía significar que se había cometido algún error o que el rotor medio o el rotor izquierdo llevaban a cabo un movimiento. Utilizando los teoremas reflejados anteriormente, mediante los cuales se obtuvo A, B, C, D, E y F a partir de DA, EB y FC , era posible determinar varias soluciones para $N^{-1}PN$ a partir de XW . También era posible obtener diversas soluciones a partir de la permutación WX , y solo existía una solución igual para XW y WX . De forma similar, era posible obtener la permutación N , correspondiente al rotor derecho, a partir de $N^{-1}PN$.

Después de llevar a cabo el razonamiento para la reconstrucción teórica de la máquina Enigma militar y de llevar a cabo numerosas pruebas, Rejewski se dio cuenta de que había cometido un error en la suposición de que la permutación H , correspondiente al cilindro de entrada, era idéntica a la de la máquina Enigma comercial. Los alemanes podían haber incluido cualquier permutación entre el gran número de posibilidades, lo que no hacía factible probar cada combinación. Sin embargo, Rejewski consiguió dar con la clave a principios de 1933 mediante la suposición de que los alemanes, destacados por su orden, estuvieran utilizando la permutación alfabética, es decir, la a se transforma en la a , la b en la b , etc. Esta suposición resultó ser correcta, logrando un hito inicial significativo.

El estudio de las relaciones entre la 1ª y 4ª, 2ª y 5ª, y 3ª y 6ª letras llevó a la conclusión de que dicha relación era una consecuencia directa de la configuración y disposición de los rotores ya que el clavijero no influía en la estructura cíclica. Rejewski denominó característica al número de ciclos y longitudes de los

rotores, que ascendía a $6 \cdot 26 \cdot 26 \cdot 26 = 105456$, un valor que reducía significativamente el número de posibilidades y permitía abordar manualmente el ataque. Rejewski recopiló todas las características en un catálogo, algo que más tarde permitiría al grupo de criptoanalistas obtener un catálogo para cada disposición de rotores. Gracias a las filtraciones de Asché, se había conseguido deducir la configuración de dos de los rotores, y a principios de 1933 se obtuvo la configuración del tercer rotor permitiendo así deducir la configuración del reflector. A finales de enero de 1933, la máquina Enigma era vulnerable.

La subida de Hitler al poder en enero de 1933 y la necesidad de refuerzos provocó que el grupo de criptoanalistas BS4 fuera reforzado con Jerzy Różycki y Henrik Zygalski en febrero de 1933. Con el objetivo de mecanizar el encuentro del catálogo adecuado, se construyeron réplicas de la Enigma militar, lo que obligó a los criptoanalistas a girar los rotores de forma manual probando todas las posibilidades. Esta dificultad condujo a la invención del Ciclómetro, una máquina Enigma doble con seis rotores y dos reflectores que permitía reducir considerablemente la rotación manual y el tiempo de obtención. Durante los años posteriores, las comunicaciones con Enigma resultaron ser un libro abierto para los polacos con una tasa de éxito superior al 75 %, teniendo en cuenta los mensajes incompletos debido a las interferencias. Esto comenzaría a cambiar a partir de 1937.

4.4.2. Hojas de Zygalski

Ante las sospechas alemanas de que sus mensajes podrían estar siendo interceptados, se tomaron medidas para cambiar la configuración de la máquina Enigma militar. En 1937, se cambió el reflector de las máquinas, lo que obligó a los polacos a reconstruir nuevamente el catálogo anterior. En 1938, se optimizó aún más la Enigma al permitir que la configuración de los rotores fuera seleccionada para cada mensaje. Esto se sumó a las claves de mensaje, lo que significaba que la cabecera ahora tendría tres letras iniciales que establecían la configuración de los rotores, siguiendo el orden diario establecido en los libros, junto con el doble cifrado de las claves de mensaje.

Estos cambios suponían la modificación de los productos DA , EB y FC en cada mensaje, pero el error alemán de repetir la clave del mensaje en cada comunicación siguió dando oportunidades a los criptoanalistas polacos. Calculando 105456 productos posibles de DA , comprobaron que el 40 % contenían ciclos de longitud 1, es decir, una letra se convertía en su idéntica. Este fenómeno también ocurría en los productos EB y FC . Algunos ejemplos son los siguientes:

FQC GBWSBM

HSW RDLRMP

DAV STLYPL

Este término de repetición fue bautizado como "female" por los británicos. En el primer caso, la cabecera se expresaba como una 2,5-female (se repetían la segunda y quinta letra) y el ciclo (B) pertenecía a la permutación EB . En el segun-

4.4. El criptoanálisis polaco de la máquina Enigma

do caso, la cabecera se expresaba como una 1,4-female (se repetían la primera y cuarta letra) y el ciclo (R) pertenecía a la permutación DA . En el último caso, la cabecera se expresaba como una 3,6-female (se repetían la tercera y sexta letra) y el ciclo (L) pertenecía a la permutación FC . Las permutaciones DA , EB y FC estaban determinadas únicamente por la posición y el orden de los rotores, lo que implicaba que el objetivo principal residía en identificar adecuadamente dichos parámetros. Había un total de 105456 configuraciones potenciales para los rotores, pero cada vez que se encontraba una "female", estas se reducían en un factor de 0.4, debido a que solo el 40% de las permutaciones DA , EB y FC presentaban ciclos unitarios. La teoría de probabilidades demostraba que al elegir al azar las permutaciones, el 11,5% de las cabeceras presentaban "females", limitando la obtención en doce o trece entre centenares de mensajes para determinar el orden y la posición de los rotores.

Sumado a los descubrimientos de Marian Rejewski, en 1938 Henrik Zygalski inventó un método para descifrar de manera masiva los mensajes cifrados obteniendo la posición y el orden de los rotores. El método se fundamentaba en la detección de females y era bastante primitivo, ya que requería 6 conjuntos (los 6 órdenes posibles de los rotores) de 26 hojas (las 26 letras iniciales posibles del rotor izquierdo). En cada hoja se asignaba una letra y se trazaba una cuadrícula de 51x51, en la cual se incluían todas las letras duplicadas desde la esquina superior izquierda. Las letras horizontales representaban las posiciones del rotor central y las verticales del rotor derecho, haciendo que cada cuadrado reflejara una permutación con ciclos de una letra correspondiente a esa posición de los rotores, una "female".

Si en dicha posición se hallaba una 1,4-female, el recuadro correspondiente sería perforado. En las 2,5-females y 3,6-females se debía llevar a cabo un proceso de normalización que consistía en adelantar una posición el Grundstellung, letra de la clave del mensaje, correspondiente a las 2,5-females o dos posiciones el correspondiente a las 3,6-females. Unos ejemplos de dicha normalización son los siguientes:

$$FQC \text{ GBWSBM} \Rightarrow FRC \text{ GBWSBM}$$

$$DAV \text{ STLYPL} \Rightarrow DAX \text{ STLYPL}$$

Una vez realizada la normalización, se repetía el proceso para cada orden de los rotores y cada posición del Ringstellung correspondiente al rotor izquierdo, primera letra de las tres incluidas en el libro de claves diarias para configurar la máquina de manera previa a elegir la clave de mensaje. A continuación, tras haber fijado el orden de los rotores, se procedía a normalizar de nuevo aquellas females cuyo Grundstellung indicara un avance del rotor central. Tomando como ejemplo la suposición de que el orden de los rotores es III II I, y el Grundstellung es FSQ, es necesario normalizar la clave del mensaje a FTQ. Esto se debe a que la letra Q ocasiona un avance en el rotor situado inmediatamente a su izquierda, es decir, el rotor central II (Basándose en una extensa investigación, se ha confirmado que el rotor I ocasionaba el avance del rotor a su izquierda cuando se alcanzaba la letra Q. De la misma manera, el rotor II provocaba el

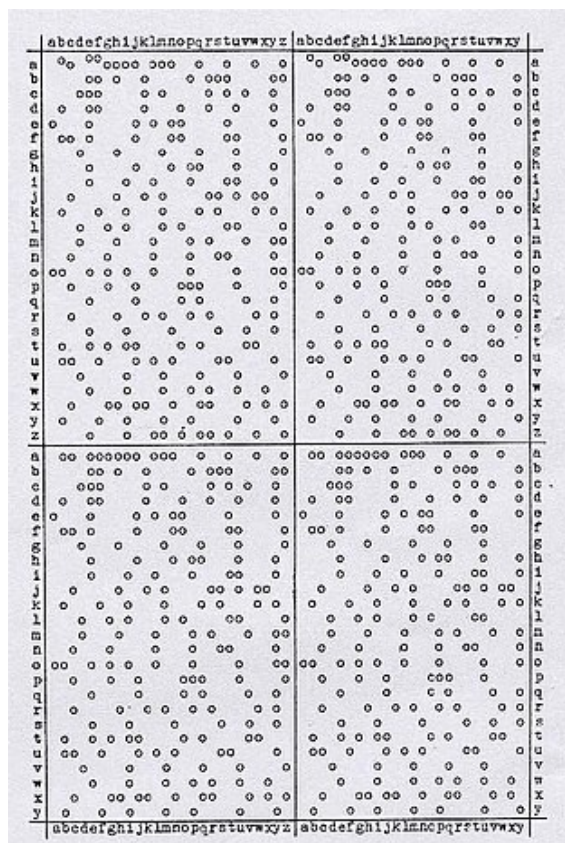


Figura 4.8: Ejemplo de hoja de Zygalski. Fuente: Wikipedia

avance al alcanzar la letra E, mientras que el rotor III lo hacía con la letra V).

El siguiente paso consistía en elegir el conjunto de 26 hojas correspondientes al orden de rotores supuesto y, fijada una letra del rotor izquierdo, considerar el Grundstellung de una primera female. Suponiendo que se tienen las 1,4-females con Grundstellung HSW y TQU, y se ha fijado la letra E, entonces primero se debe escoger la hoja C, ya que $H-E=C$, que servía como patrón básico para empezar a trabajar colocándose sobre una mesa transparente iluminada por debajo. Seguidamente se seleccionaba la hoja correspondiente a la letra O, ya que $T-E=O$, pero desplazada 2 recuadros hacia la derecha (de la Q a la S van 2 letras y el eje horizontal representaba las posiciones del rotor central) y 2 recuadros hacia abajo (de la U a W van 2 letras y el eje vertical representaba las posiciones del rotor derecho), sobre la posición CC. Tras repetir la operación con todas las females encontradas había que observar si la luz de la mesa atravesaba una única perforación, la cual proporcionaba de manera inmediata el orden de los rotores y el Ringstellung del rotor izquierdo. Para determinar la posición del rotor central y derecho bastaba con calcular la diferencia entre el Grundstellung de la female y las letras cifradas de la perforación. Esto se debía a que fijada una de las females, las letras de la perforación correspondiente determinaban la posición de los rotores en el momento del cifrado, equivalente a la diferencia entre el Grundstellung de la female y el Ringstellung que se pretendía obtener. En situaciones

4.4. El criptoanálisis polaco de la máquina Enigma

donde la luz atravesaba más de una perforación, era necesario repetir el proceso de cálculo de la diferencia mencionado anteriormente para cada caso posible. Este proceso se realizaba hasta que las contradicciones resultantes permitían descartar todas las posibles soluciones excepto la correcta.

Tras haber descubierto el orden y la posición de los rotores, la última operación era la obtención de las conexiones del Stecker o clavijero. Anteriormente se demostró que el clavijero no alteraba la estructura de ciclos, sino que solamente alteraba las letras de los mismos, en este caso los ciclos de longitud 1 del catálogo de características (letras que se convertían en ellas mismas) por las letras repetidas de las females. Esto significaba que la letra repetida de una female estaba conectada con una de las letras pertenecientes a un ciclo de longitud 1 en la permutación DA . Al contemplar todas las females interceptadas se llegaba a la deducción de conexiones sin mucha dificultad.

4.4.3. Método del reloj

El método del reloj desarrollado por Jerzy Różycki fue otra de las contribuciones significativas de los polacos a la lucha contra Enigma. Este método permitía determinar en ciertas ocasiones el rotor que estaba en la posición del rotor derecho o rotor rápido. A medida que los alemanes aumentaron la frecuencia con la que cambiaban el orden de los rotores en sus máquinas Enigma, pasando de cada tres meses a diariamente a partir de 1936, la capacidad de obtener la disposición del rotor rápido se volvió aún más crucial en los esfuerzos para descifrar los mensajes encriptados. Una de las distinciones de este método radicaba en su fundamentación en la frecuencia de aparición de las letras del alfabeto alemán, uno de los métodos más utilizados en el criptoanálisis que no había sido utilizado por los polacos previamente.

Al situar dos textos en alemán uno debajo del otro es posible apreciar que, de media, la probabilidad de coincidencia de dos letras era de $2/23$. Un ejemplo extraído de [4] es el siguiente:

W E M G O T T W I L L R E C H T E G U N S T ...

D E R A L T E L A N D M A N N A N S E I N E N ...

Se sabía que este hecho se repetiría para el texto cifrado con una clave idéntica, pero su cifrado mediante la máquina Enigma con distinta configuración era la siguiente:

V D Z T H D B G H X S P V Y E C G F I A D H ...

F B X G G P A X H W X U O F M Q H U U B Z ...

La probabilidad de coincidencia de una letra en dos textos cifrados con claves diferentes era de $1/23$, debido a la frecuencia de aparición de letras en el idioma alemán. En textos de 23 letras, la repetición era poco frecuente, pero en mensajes más largos, como de 260 letras, era posible determinar si se había utilizado la misma clave para llevar a cabo el cifrado. Por tanto, con una cantidad suficiente de texto disponible era posible encontrar alrededor de una docena de pares de

mensajes en los que en cada pareja las dos primeras letras de las claves eran idénticas y la tercera era diferente. Este tipo de mensajes se superponía haciendo uso de una de las dos formas de superposición dependientes de la posición de partida del rotor rápido al desplazarse el rotor medio. Estas posiciones eran conocidas y diferentes para cada uno de los tres rotores, el rotor I avanzaba al desplazarse de la letra Q a la R, el rotor II al desplazarse de la E a la F y el rotor III al desplazarse de la V a la W. Para cada una de las 2 formas de expresar los mensajes era suficiente con contar el número de columnas con idénticas letras y así determinar el modo correcto de escritura junto con el rotor posicionado en el lado derecho.

4.4.4. Método ANX

Otro de los métodos que permitían obtener el cifrado de la Enigma era el método ANX. Muchos de los textos claros de los mensajes alemanes interceptados comenzaban con las letras ANX ya que en alemán "AN" significa "para" y la "X" se empleaba comúnmente como separador de palabras. Al obtener la posición inicial del rotor derecho en un mensaje comenzado por ANX, el resto podía obtenerse gracias al catálogo de características desarrollado por Rejewski.

4.4.5. Bombas

El grupo de criptoanalistas polacos había fabricado réplicas de la máquina Enigma militar con el propósito de agilizar el proceso de descifrado de mensajes. Marian Rejewski, con dichos conocimientos, optó por crear un dispositivo electro-mecánico cuya idea combinaba seis de estas réplicas para obtener las claves diarias utilizadas por los alemanes en el mismo día. Esta máquina, bautizada como bomba por los franceses unos años más tarde, era capaz de mecanizar el sistema de catalogación al probar 17576 combinaciones en menos de dos horas.

El funcionamiento de la bomba se basaba en la intercepción de tres females con la repetición de una misma letra, un ejemplo es el siguiente:

HSW LPTLRA PVN RLSWLP KCB GLWBLX

Al tomar la primera female se afirmaba que el ciclo (L) estaba presente en la permutación DA (por ser una 1,4-female). El número de productos DA era de 105456, número que representaba las posibles configuraciones de los rotores de los cuales se desconocían cuántos contenían el ciclo (L). Con los conocimientos de que el 4 % de los productos DA contenían dicho ciclo, las 105456 posibilidades se reducían en un factor de 0,04 cada vez que se observase una female con la letra L. En este caso de ejemplo $105456 \cdot (0,04)^3 = 6,749$, por lo que existían unas 7 configuraciones que ocasionaban la repetición de las tres females. Las bombas eran las encargadas de automatizar el proceso de identificación de dichas configuraciones. Cada bomba consistía en tres ciclómetros conectados, un motor eléctrico accionaba los seis rotores de forma sincronizada recorriendo las 17576 posiciones posibles. Al alcanzar una posición en la que los ciclómetros reconocían el ciclo programado, la máquina se detenía mostrando la posición.

4.4. El criptoanálisis polaco de la máquina Enigma

Ante las $3! = 6$ posibles disposiciones de rotores, los criptoanalistas necesitaban tener seis máquinas, cada una con una disposición, trabajando en paralelo.

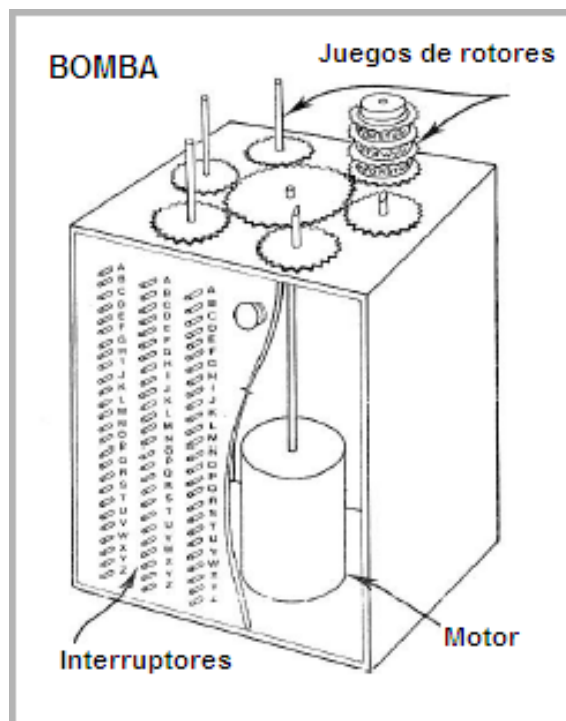


Figura 4.9: Diseño original de una bomba de Rejewski. Fuente: Mikel García Larragán [21]

Una vez obtenida la configuración, quedaba por determinar las conexiones del clavijero. Para ello, se retiraban todos los cables del clavijero de las réplicas y se procesaban todos los mensajes interceptados. En ciertas ocasiones, se encontraban mensajes como "DEBEMON ATACAR PARIN", que permitían deducir la conexión del clavijero entre las letras "S" y "N". Con un gran número de mensajes interceptados, era posible deducir todas las conexiones del clavijero de manera sistemática. Las bombas permitieron la reducción de los errores humanos, pero la necesidad de encontrar tres females con letras idénticas que permanecieran invariantes por el clavijero no era una tarea fácil, lo que provocó que no fuera un método de uso diario.

El trabajo del BS4 mantuvo al gobierno polaco al tanto de los mensajes interceptados por los alemanes y de las pretensiones de invasión, pero los aliados no tomaron en consideración dichos avisos. En diciembre de 1938, las Enigmas utilizadas por el ejército alemán fueron dotadas con dos nuevos rotores que aumentaban el número de bombas necesarias a 60 (variaciones de 5 elementos tomados de 3 en 3), algo para lo que los polacos no tenían recursos. Se continuaron interceptando y descifrando ciertos mensajes de manera manual después de deducir el cableado interno de los nuevos rotores a partir del incremento en las permutaciones, aunque esto no resultó ser suficiente. Por si fuera poco, en enero

de 1939 los alemanes ampliaron las conexiones del clavijero a 10, dejando totalmente obsoleto el método de las bombas. Ante el aislamiento, la inteligencia polaca decidió compartir sus conocimientos con los aliados con la esperanza de que sus recursos fueran suficientes para poner fin a la inviolabilidad de Enigma, que volvía a resurgir.

4.5. Bletchley Park

En julio de 1939, ante la inminente invasión de Polonia por parte del ejército alemán y la falta de recursos, el Biuro Szyfrów decidió compartir con los aliados todos sus conocimientos sobre el descifrado de Enigma. El jefe de los criptoanalistas británicos de la Foreign Office, Alfred Dillwyn Knox, organizó una reunión entre británicos, franceses y polacos en el viejo búnker en el que había estado trabajando el BS4 en Pyry, cerca de Varsovia. Entre los británicos, estuvo presente el comandante Alastair Denniston, quien lideraba las operaciones criptográficas en la Estación X (Bletchley Park) y era un ferviente defensor de la relevancia de las matemáticas en la lucha contra la máquina Enigma. Por parte de los franceses, asistieron Gustave Bertrand y el capitán Henri Braquenié. Mientras tanto, los representantes de Polonia fueron el capitán Maksymilian Ciężki, el teniente coronel Gwido Langer y el coronel Jefe Stefan Mayer. Después de la reunión, los aliados fueron puestos al tanto de todos los avances logrados por el grupo de criptoanalistas BS4 y recibieron réplicas de las bombas inventadas por Marian Rejewski. El 1 de septiembre de 1939, Adolf Hitler invadió Polonia, marcando el inicio de la Segunda Guerra Mundial.

Desde la aparición de la máquina Enigma, los aliados creían que era imposible de descifrar, pero el conocimiento obtenido de los criptoanalistas polacos manifestó la importancia de los matemáticos en este proceso. Bletchley Park, sede del Government Code & Cypher School, se convirtió en el centro secreto del ataque a Enigma tras el estallido de la guerra. Durante el otoño de 1939, los matemáticos, criptólogos y científicos de Bletchley se dedicaron a estudiar y dominar las técnicas del BS4 a medida que el personal se iba incrementando.

El primer método que utilizaron los británicos en su lucha contra Enigma fue el de las hojas de Zygaliski, organizado por el matemático R. F. Jeffreys del Downing College de Cambridge. Para finales de 1939, los trabajadores de Bletchley Park ya tenían listas las 1560 hojas necesarias (26 para cada letra multiplicado por las 60 nuevas combinaciones de rotores). Se realizaron pequeñas modificaciones por lo que fueron bautizadas como hojas de Jeffreys, a pesar de que su funcionamiento era casi idéntico al método original ideado por el criptoanalista polaco. Simultáneamente, parte del equipo se dedicó a analizar el tráfico de mensajes alemán para identificar las distintas redes de comunicaciones, ya que cada una operaba con una clave diferente. La gran cantidad de recursos que requería el método de las hojas de Zygaliski obligó a los británicos a reclutar matemáticos de Cambridge, alumnos de últimos cursos, jugadores de ajedrez o lingüistas. La aplicación de este método hizo percatarse a los británicos que los operadores alemanes utilizaban a menudo claves realmente obvias, producto del agotamiento en los campos de batalla. Algunos ejemplos de estas claves eran QWE o FGH,

situadas de manera consecutiva en el teclado de la máquina Enigma de 4.10.



Figura 4.10: Disposición de letras original de una máquina Enigma. Fuente: Microservos [22]

Estas debilidades, similares a las acontecidas en la Primera Guerra Mundial por parte del bando alemán, fueron denominadas *cillis*. Además de las *cillis*, también se encontraron otro tipo de debilidades, como la prohibición de que un mismo rotor no podía permanecer en el mismo hueco más de un día, lo que provocó la simplificación del trabajo de los criptoanalistas al reducir las posibilidades. Todas estas fragilidades eran factores que los británicos no podían desaprovechar.

4.6. Alan Turing

Alan Turing, el genio matemático del King's College de Cambridge, es considerado la figura más importante en el ataque a la máquina Enigma, recibiendo el apodo de "El hombre que descifró el código Nazi". Antes de unirse a Bletchley, Turing había trabajado en el desarrollo del concepto de la máquina computacional creando la llamada máquina de Turing, una máquina virtual o física que permitía definir el concepto de algoritmo.

En Bletchley Park, Turing estuvo a cargo del barracón 8, encargado de descifrar los códigos de la Enigma utilizados por la marina con el objetivo de derrotar a los potentes submarinos alemanes. Las máquinas utilizadas por la marina (denominadas SHARK) eran prácticamente impenetrables, ya que contaban con un rotor adicional y sus operadores eran extremadamente cuidadosos debido a las importantes consecuencias que podría tener la obtención del texto claro por parte del enemigo. Además de esto, los alemanes se percataron del compromiso

Capítulo 4. Criptografía en la Segunda Guerra Mundial

en la seguridad que suponía la repetición de la clave al interceptar diversas hojas de Jeffreys y documentos que reflejaban el funcionamiento del método. Por lo tanto, el 10 de mayo de 1940, coincidiendo con la toma de París, modificaron el sistema de cifrado eliminando la repetición de la clave. El cambio supuso una situación crítica para los aliados, las females de las cabeceras desaparecían y las hojas de Jeffreys quedaban inutilizables.



Figura 4.11: Alan Turing. Fuente: ¿Puede pensar una máquina? [23]

Turing ya había estado buscando una forma de ataque que no dependiera de la repetición de la clave. Empezó basándose en los *puntales* (cribs en inglés), asociaciones de texto claro con el texto cifrado. Después de una exhaustiva investigación, lograba predecir con precisión partes del mensaje, aprovechando el conocimiento de la hora y el lugar de emisión. Un ejemplo de estos *puntales* se observaba en los mensajes cifrados temprano por la mañana por los alemanes, que a menudo informaban sobre la meteorología y probablemente contenían la palabra "wetter", que significa "tiempo" en alemán. Dado el diseño de la Enigma, se empezaron a buscar textos con la misma longitud donde ninguna letra de la palabra "wetter" coincidiera consigo misma después del cifrado, ya que ninguna letra se cifraba en ella misma en la máquina. Aunque parecían simples suposiciones y no era viable atacar textos por este método, Turing tenía la certeza de que era el camino correcto para desarrollar el método definitivo.

Se implementó una estrategia similar a la utilizada por Marian Rejewski, que consistía en separar los efectos de cada uno de los componentes de la máquina. Se analizaban por separado los rotores (tanto su disposición como su configu-

ración) y las conexiones del clavijero. Esta metodología permitía, por ejemplo, al descubrir un detalle en un *puntal* que no guardaba relación con las conexiones del clavijero, probar las $60 \cdot 17576 = 1054560$ combinaciones posibles. Además, una vez descubiertas las posiciones de los rotores, era posible deducir las conexiones del clavijero. Se comenzaba estudiando los rizados internos, un ejemplo en el que se había estimado que la palabra *wetter* había sido cifrada con ETJWPX puede observarse en la Figura 4.12.

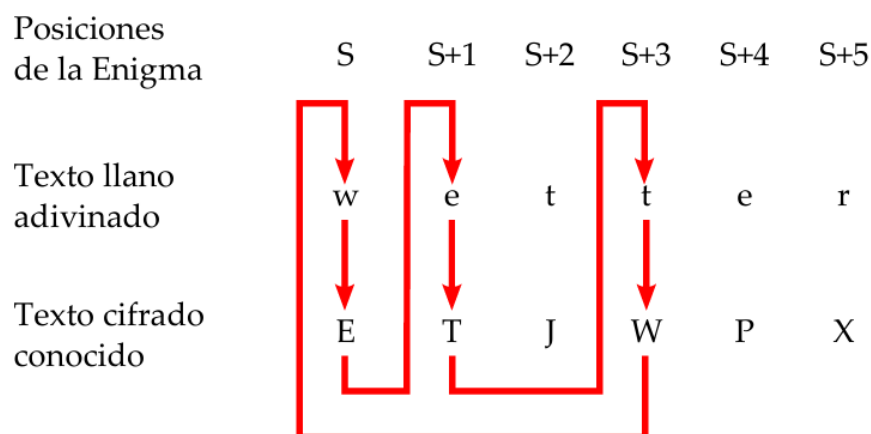


Figura 4.12: Rizos internos del puntal "wetter". Fuente: Criptología Nazi. Los Códigos Secretos de Hitler [4]

En la Figura 4.12, cuando la Enigma estaba en la posición S, la letra "w" se transformaba en la letra "E"; en la posición S+1, la letra "e" se transformaba en la letra "T", y así sucesivamente. Basándose en el funcionamiento de las bombas polacas, Turing pudo desarrollar un modelo mejorado, conocido como "bombe", cuya función era probar numerosas posibilidades utilizando la configuración inicial de rizados establecidos por los *puntales*. Además del desarrollo del dispositivo, Turing también diseñó varios procedimientos estadísticos para mejorar la eficiencia de la "bombe", los cuales fueron llamados *bamburismos* debido a que las láminas utilizadas para llevar a cabo dichos procedimientos eran fabricadas en una fábrica ubicada en Banbury, una región británica. El primer modelo, llamado Victory, fue instalado en Bletchley Park en marzo de 1940, pero al tratarse de un dispositivo en fase de desarrollo, presentaba algunos errores y problemas.

William Gordon Welchman, también matemático y con una gran mente como ingeniero, fue capaz de encontrar la solución a muchos de los problemas y de saber cómo implementar muchas de las ideas que surgían en el equipo de criptoanalistas. En agosto de 1940 se puso en marcha el modelo Spider, incluía el panel diagonal de Welchman que aprovechaba la reciprocidad de letras en el clavijero, es decir, si una letra *X* estaba permutada por *Y*, entonces *Y* estaba permutada por *X*. Las "bombe" estaban formadas por 90 rotores con el mismo cableado interno que la máquina Enigma y, además, el reflector era simulado gracias a que las conexiones y cables estaban duplicados. Su funcionamiento se basaba en un procedimiento de prueba y error entre todas las posibles combinaciones

Capítulo 4. Criptografía en la Segunda Guerra Mundial

de los rotores: si el resultado era una contradicción entonces se descartaba dicha posición y, en caso contrario, se seleccionaba la solución como candidata. Las "bombe" eran capaces de identificar las palabras en los textos cifrados con una gran probabilidad mediante el uso de los *puntales*, lo que provocó que el gobierno británico, por medio de Winston Churchill, aumentase la financiación de Bletchley Park.

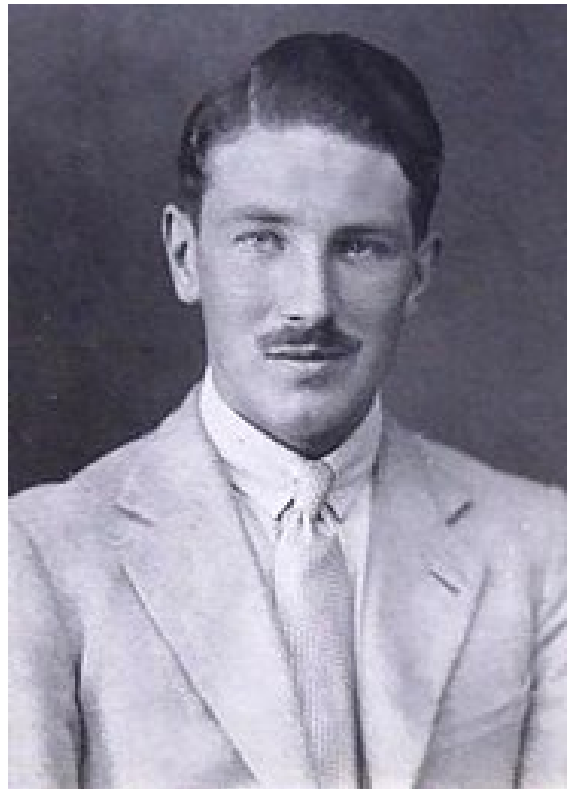


Figura 4.13: William Gordon Welchman. Fuente: Wikipedia [24]

Ante las sospechas alemanas de que sus mensajes estaban siendo descifrados por los criptoanalistas británicos, a finales de 1941 se introdujo un nuevo espacio para rotor en la máquina Enigma y nuevos rotores con un cableado interno desconocido. Ahora la máquina utilizaba cuatro rotores y las posibles combinaciones pasaban a ser de 1680 (Variaciones de 8 elementos tomados de 4 en 4). Sin embargo, otro error humano ocurrió y volvió a comprometer la seguridad alemana. En este caso, uno de los submarinos emitió un mensaje utilizando una Enigma de tres rotores, lo que facilitó establecer la correspondencia entre los dos sistemas y obtener el cableado del nuevo rotor. Para finales de 1942, Bletchley Park ya había sido capaz de romper el nuevo cifrado.

En los últimos meses de la guerra, había 211 máquinas "bombe" en Bletchley Park ayudando a que los aliados ganaran la guerra. Algunos de los eventos más importantes en los que la descryptación de la Enigma gracias a la máquina de Turing jugó un papel crucial fueron la batalla del Atlántico o el desembarco de Normandía. Tras lograr la victoria, Winston Churchill ordenó destruir todas las máquinas "bombe" y toda la documentación relacionada con las actividades

que se llevaban a cabo en Bletchley Park ante la inminente Guerra Fría que iba a suceder. No sería hasta 1976, después de que Alan Turing falleciera en 1952, habiéndose suicidado tras ser sometido a castración química debido a su homosexualidad, que todas sus actividades serían reveladas y recibiría el reconocimiento que merecía.

4.7. Las máquinas Lorenz

En 1940, los británicos empezaron a interceptar comunicaciones enviadas a través de teletipos que no empleaban el código Morse, el cual era utilizado por los operadores para transmitir los mensajes cifrados mediante la máquina Enigma. Estas comunicaciones estaban cifradas utilizando las máquinas Lorenz SZ40 y SZ42, las cuales eran utilizadas por Hitler y otros altos mandos alemanes debido a su mayor sofisticación y complejidad. De esta manera, los alemanes reservaban la máquina Enigma para el uso en los campos de batalla, mientras que la máquina Lorenz se destinaba a la comunicación entre los altos cargos.

La tecnología del teletipo posibilitaba el envío directo de mensajes, lo que disminuía la necesidad de recursos humanos en comparación con la máquina Enigma. Mientras que la Enigma requería un mínimo de tres individuos en el lugar de emisión y otro tanto en el de recepción (un operador encargado de teclear el mensaje, otro de registrar las letras del panel luminoso que se iluminaban y un operador de radio encargado de enviar o recibir el mensaje), las máquinas Lorenz solo precisaban un operador en el sitio de emisión y otro en el de recepción.

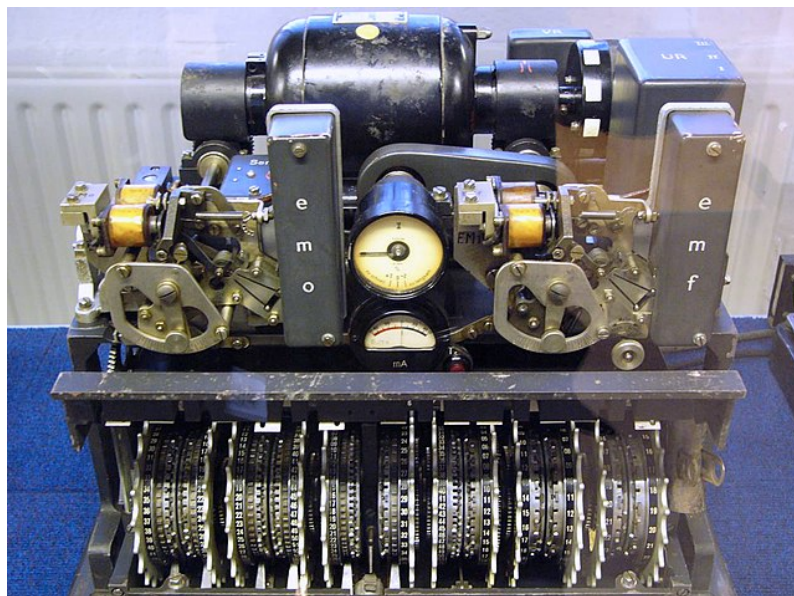


Figura 4.14: Máquina Lorenz. Fuente: Wikipedia [25]

Las máquinas Lorenz estaban equipadas con 12 rotores y utilizaban el código Baudot, en el que cada letra o símbolo estaba representado por 5 pulsos que podían indicar la ausencia (o) o presencia (•) del mismo. El proceso de encriptación

implicaba la generación de una secuencia aleatoria de 5 pulsos y la operación de cada pulso de la letra a cifrar con el pulso correspondiente de la secuencia aleatoria mediante la operación XOR, establecida por Vernam en su cifrado en 1919. Los 12 rotores, conocidos como pinwheels, eran responsables de generar esta secuencia aleatoria. De estos, cinco se denominaban χ y giraban paso a paso de manera regular, otros cinco se denominaban ψ y giraban paso a paso de forma irregular, mientras que las dos últimas eran ruedas motoras y se identificaban como μ_1 y μ_2 . Un ejemplo de cifrado para la letra D, representada como $\bullet \circ \circ \bullet \circ$, es el siguiente:

1. El operador de la máquina Lorenz codificaba los rotores χ , para este ejemplo se supone que se seleccionan on-off-off-on-on, lo que significa que se aplicaba la operación XOR de Vernam, también conocida en este ámbito como suma, a la secuencia que representaba la letra original y a la secuencia modificada por los rotores χ . En este ejemplo se haría $D + \chi = \bullet \circ \circ \bullet \circ + \bullet \circ \circ \bullet \bullet = \circ \circ \circ \circ \bullet$
2. El siguiente paso que seguía la máquina era la aplicación de los rotores ψ . Estos rotores se movían en función de la posición de los rotores μ_1 y μ_2 , creando una secuencia aleatoria. Para el ejemplo se supone que la secuencia de los rotores ψ era off-on-on-on-off. Por tanto, a la letra $\circ \circ \circ \circ \bullet$ se le sumaba $\circ \bullet \bullet \circ \bullet$ dando como resultado $\circ \bullet \bullet \bullet \bullet$, es decir, $00001 + 01110 = 01111$

Por tanto, la letra inicial introducida, la letra D representada como $\bullet \circ \circ \bullet \circ$, se sustituye por la letra V, representada por $\circ \bullet \bullet \bullet \bullet$. Como los pinwheels sufren rotaciones por cada letra cifrada, si se vuelve a introducir la letra D el resultado obtenido seguramente sea distinto ya que la secuencia aleatoria no sería la misma. Como la máquina era recíproca, bastaba con configurar la máquina inicialmente de la misma forma que el emisor para obtener el texto claro a partir del texto cifrado.

Los criptoanalistas de Bletchley Park denominaron esta máquina como Tunny, ya que asignaban códigos o claves para referirse a las diversas redes de comunicación alemanas, dado el carácter confidencial de su labor. Al igual que con la máquina Enigma, los altos mandos alemanes incluían una clave de mensaje compuesta por secuencias de 12 caracteres, conocidas como indicadores, los cuales representaban cada uno de los 12 rotores de la máquina Lorenz. Estos indicadores solían adoptar la forma de letras, como por ejemplo (GMIL-SEOCNVDF), o nombres propios como Geoffrey, Mike, Ilya, Ludwig, correspondiendo al indicador (GMIL...). Asimismo, se les proporcionaba un libro de códigos mensual que contenía las equivalencias de cada letra para cada día del mes. Esto se debía a que, en ocasiones, la letra L de un indicador representaba la configuración de la letra S en el rotor, mientras que en otros días representaba la introducción de la letra A. [4]

4.7.1. Criptoanálisis y descifrado

El 30 de agosto de 1941, se interceptó un mensaje cifrado de alrededor de 4000 caracteres, lo cual era considerablemente grande en comparación con la mayo-

ría de los mensajes habituales. El operador alemán a cargo del envío recibió una solicitud de retransmisión debido a que el mensaje original no había sido recibido correctamente. Cometió un error crucial al reenviar el mensaje con el mismo indicador de la máquina Lorenz que se había utilizado en el primer intento, a pesar de haber abreviado algunas palabras, rompiendo una de las normas básicas de la criptografía. Este error manifestó una vulnerabilidad que el equipo de criptoanalistas de Bletchley Park, liderado por John Tiltman, intentaría explotar.

Tiltman ya había trabajado anteriormente con otras cifras alemanas y otras cifras japonesas por lo que estaba familiarizado con el criptoanálisis. Comenzó su análisis estudiando las propiedades del grupo $(\mathbb{Z}_2, +_2)^5 = (\{0,1\}^5, XOR)$ y observando que si se aplicaba la operación XOR a dos letras, se toman como ejemplo L (○ ● ○ ○ ●) y R (○ ● ○ ● ○) obteniendo O (○ ○ ○ ● ●), entonces al aplicar nuevamente a O (○ ○ ○ ● ●) la operación con R (○ ● ○ ● ○) se obtenía L (○ ● ○ ○ ●), es decir, la letra original.

$$\text{○ ● ○ ○ ●} + \text{○ ● ○ ● ○} = \text{○ ○ ○ ● ●} \Rightarrow \text{○ ○ ○ ● ●} + \text{○ ● ○ ● ○} = \text{○ ● ○ ○ ●}$$

Tras aplicar la operación XOR (también llamada suma) a los dos textos cifrados, se obtuvo una cadena de caracteres que representaba la suma de los textos claros, ya que denotando como P_1 y P_2 a los dos textos planos y sumando la secuencia K para obtener los dos textos cifrados C_1 y C_2 ($P_i + K = C_i, i = 1, 2$), entonces al sumar los dos textos cifrados se obtenía la suma de los dos textos planos utilizando las propiedades del grupo $C_1 + C_2 = (P_1 + K) + (P_2 + K)$ con $K + K = 0$ (considerando a 0 como el elemento neutro de esta aritmética, que sería equivalente a ○ ○ ○ ○ ○). Si se realizaba la suma de dos textos cifrados de forma idéntica, es decir, con la misma clave, el resultado era una secuencia de elementos neutros (○ ○ ○ ○ ○), por lo que gracias a las pequeñas modificaciones que realizó el operador se pudo dar comienzo al ataque. Al haber eliminado la secuencia de cifrado, Tiltman se enfocó en que cada carácter del texto era la suma de otros dos caracteres que en el mensaje original estaban pocas posiciones, lo que inició una fase "adivinatoria" para descubrir qué palabras, al ser sumadas con el texto obtenido, revelaban el mensaje original. En el verano de 1941, Tiltman ya había descifrado el cifrado Vernam siguiendo la estrategia de que sumando el texto plano a cualquiera de los dos mensajes cifrados interceptados, se obtenía la secuencia de caracteres utilizada como clave. Todo el equipo de Bletchley Park empezó a trabajar con la esperanza de encontrar la lógica que explicara la aparente aleatoriedad de las letras obtenidas con la Lorenz.

Todos los miembros del equipo confiaban en encontrar una sistematización similar a la que habían sido capaces de obtener con la máquina Enigma, pero el principal inconveniente era que la información sobre la máquina era escasa. Ante esta situación de adversidad y tras mucho tiempo invertido, se decidió acudir a un brillante estudiante de química de la Universidad de Cambridge que había sido previamente rechazado en Bletchley Park, Bill Tutte. Tutte tenía experiencia previa en el criptoanálisis con el código Hagelin y la máquina sueca C-36, lo que decantó a Gerry Morgan, director de investigación del departamento de Tiltman, a contratarle para el equipo. El estudiante, que estaba muy familiarizado con

el método de Kasiski, empezó a buscar patrones de repetición y comportamientos periódicos de la clave. Su primera idea fue colocar los caracteres cifrados en código Baudot en vertical en 7 filas de 575 letras cada una (la cantidad de 575 se debía a que al colocar todos los caracteres cifrados en un rectángulo se observaban ciertas repeticiones cada 23 y más adelante cada 25 posiciones, haciendo $23 \cdot 25 = 575$) llegando a observar que existían algunas repeticiones en una diagonal y que obtendría mejores resultados al utilizar un periodo de 574. Realizó un nuevo intento y logró observar numerosas repeticiones de patrones de longitud 5 o 6, por lo que decidió reducir el periodo a 41 (factor primo de 574), mejorando los resultados. Siguiendo estos pasos, fue descubriendo el funcionamiento de las máquinas Lorenz, algo sorprendente teniendo en cuenta que no había visualizado una máquina previamente.

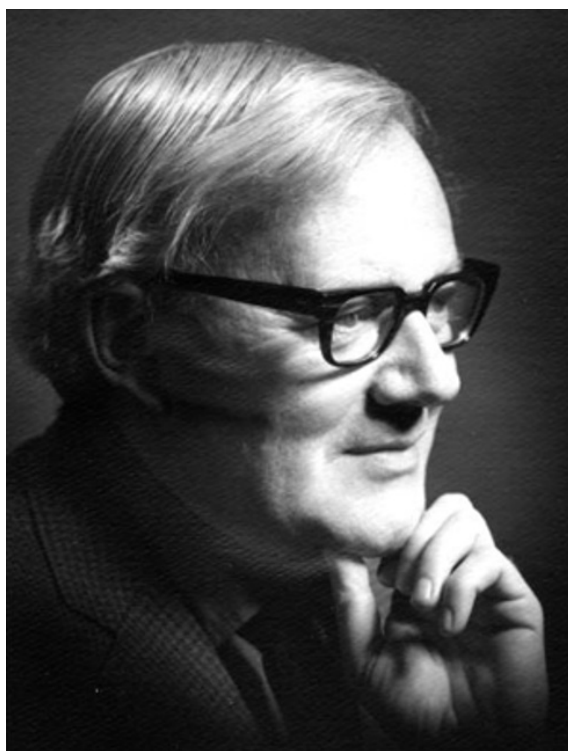


Figura 4.15: William Thomas "Bill" Tutte. Fuente: Cambridge Mathematics [26]

La máquina Lorenz hacía uso de un algoritmo para la obtención de números pseudoaleatorios, algo muy habitual en los ordenadores modernos para simular la obtención de números aleatorios. Cada uno de los rotores o pinwheels poseía varias posiciones en su periferia con un pin que admitía las posiciones on y off, afectando o no durante la rotación a otras partes de la máquina. El número de pins era variable en cada rueda, $\chi_1, \chi_2, \chi_3, \chi_4$ y χ_5 tenían 41, 31, 29, 26 y 23 respectivamente, mientras que $\psi_1, \psi_2, \psi_3, \psi_4$ y ψ_5 presentaban 43, 47, 51, 53 y 59 cada una de ellas, y finalmente las dos ruedas motoras μ_1 y μ_2 contaban con 37 y 61 respectivamente. Mientras los rotores χ giraban todas una posición por carácter, los rotores ψ giraban todas a la vez pero no con cada carácter ya que su movimiento estaba controlado por las dos ruedas μ . En el modelo SZ40, la rueda

μ_1 giraba una posición por carácter, pero la rueda μ_2 solo giraba cuando el pin de la periferia estaba en on, haciendo que girasen todos los rotores ψ . El número de pines en cada rotor se trataba de un número primo relativo al del resto de rotores para lograr el máximo periodo en el patrón de repetición, eran 501 pines que suponían $2^{501} \approx 10^{151}$ posibilidades de cifrado. A pesar del número tan alto de posibilidades de cifrado, el tomar los cinco impulsos de manera independiente provocaba números resultantes mucho más manejables.

Tutte había desarrollado un método manual para obtener la configuración inicial de las máquinas Lorenz, pero resultó ser ineficaz, dado que el descifrado del mensaje podía tardar semanas, volviendo así inútil la obtención del texto claro al referirse a órdenes militares obsoletas para cuando se lograra el descifrado. Este hecho motivó a buscar un nuevo método que fuera capaz de obtener la gran mayoría de caracteres del mensaje cifrado y así poder aportar una ventaja significativa para las tropas británicas. En noviembre de 1942, Tutte desarrolló un método, bautizado como Método Estadístico, capaz de descifrar el mensaje en gran proporción. Se llevaban a cabo cálculos mediante la comparación de dos secuencias de caracteres en código Baudot, haciendo un recuento del número de veces que cada uno tenía un pulso o una ausencia. [4]

4.7.2. Colossus

Una vez formalizado y con el objetivo de automatizar el método de descifrado de la máquina Lorenz, Tutte se puso en contacto con Max Newman, jefe de la sección de desarrollo mecánico en Bletchley Park, sugiriéndole la creación de una nueva máquina similar a las bombe que habían surgido para el ataque a Enigma. Newman y un equipo de técnicos de Bletchley se encargaron del desarrollo y construcción de máquinas optomecánicas poco eficientes con dos cintas para automatizar el método de Tutte conocidas como las Robinson. Al intentar aumentar la velocidad de lectura de datos por encima de los 1000 caracteres por segundo, se producía un estiramiento excesivo en las cintas que daba lugar a errores. Siguiendo la misma idea del equipo de Newman, Tommy Flowers, ingeniero de la British Post Office Research Station en Dollis Hill, fue el encargado de desarrollar un nuevo prototipo al que bautizó como Colossus. En el Colossus se sustituían los relés tradicionales por circuitos electrónicos a base de válvulas que permitían aumentar la velocidad de lectura hasta los 5000 caracteres por segundo.

El ordenador Colossus estaba equipado con 1500 válvulas, tiratrones y fotomultiplicadores. Las válvulas eran los antecesores de los actuales diodos o transistores, los tiratrones eran tubos rellenos de gas neón o xenón que funcionaban como un rectificador para grabar 1 bit (la combinación de ellos creaba el registro de desplazamiento), y los fotomultiplicadores eran los encargados de detectar la luz. Aplicando tiratrones y fotomultiplicadores, Colossus era capaz de leer los caracteres de la cinta aplicando una función lógica programada para cada carácter. Si el resultado de aplicar la función era correcto, se continuaba leyendo la cinta; en caso contrario, se descartaba la opción y se finalizaba el análisis.

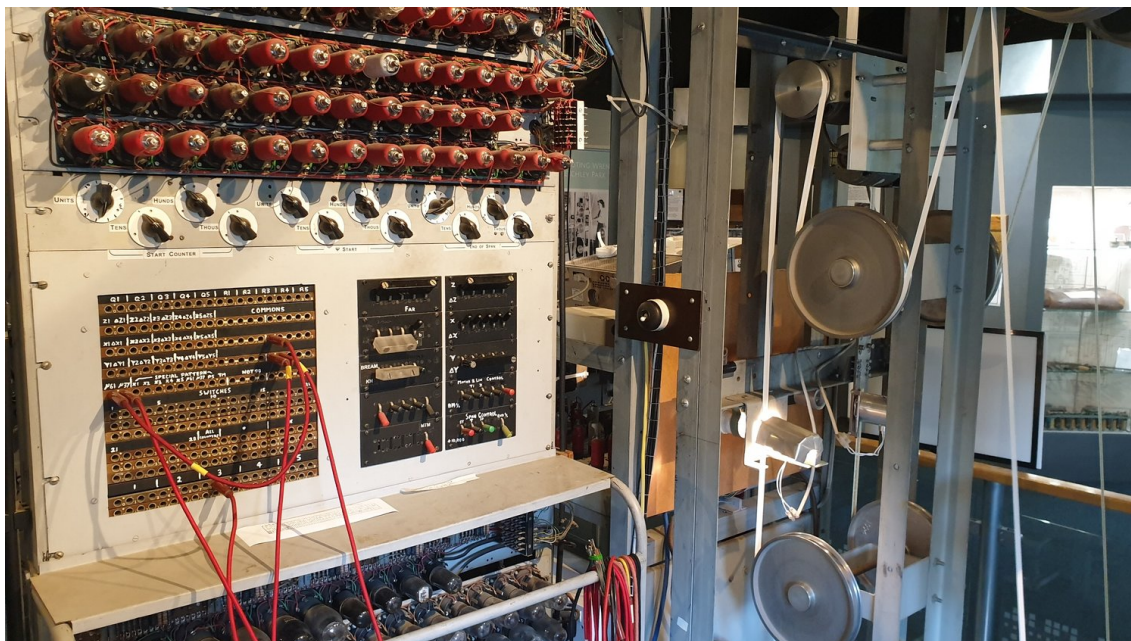


Figura 4.16: Colossus. Fuente: The National Museum of Computing [27]

En enero de 1944, se construyó la primera versión de Colossus, conocida como Mark I. En junio de 1944, se oficializó la segunda versión, conocida como Mark II. Al finalizar la guerra, se habían construido un total de 10 máquinas Mark II en Bletchley Park. Sin embargo, al igual que con las bombe, Winston Churchill ordenó su destrucción al finalizar la guerra. Esta decisión del primer ministro británico provocó que los creadores de Colossus nunca fueran reconocidos por haber construido la primera computadora de la historia, dejando todo el mérito a los creadores de la máquina ENIAC en 1946. [4]

4.8. Criptografía y computación

A pesar de la destrucción y de la obligación de mantener en secreto todo lo realizado en Bletchley Park, los conocimientos adquiridos durante la Segunda Guerra Mundial en el mundo de la computación fueron vitales para el futuro de la criptografía. La computación permitía que se crearan nuevas formas de cifrar mensajes y de atacarlos gracias a la velocidad de cálculo y a la posibilidad de trabajar con procesos repetitivos, complejos y con una gran cantidad de información. Esto provocó que se diseñaran sistemas con una seguridad muy alta aprovechando la importancia de aspectos de las claves como su aleatoriedad y su longitud.

A medida que fueron avanzando los años, el escenario era cada vez más prometededor. En los años sesenta y setenta, la computación se fue haciendo más popular a medida que mejoraban las prestaciones y se bajaban los precios. Fue en 1965 cuando Gordon Moore, cofundador de Intel, enunció la conocida Ley de Moore que establecía que cada dos años se duplicaba el número de transistores en los microprocesadores aumentando la potencia de los mismos. El pronóstico de

Moore se ha cumplido en cierta medida hasta el día de hoy, demostrando que no estaba muy equivocado con su afirmación.

Cuando los ordenadores ya estaban lo suficientemente extendidos, se empezó a hablar de estándares públicos de criptografía para proteger las comunicaciones y la información almacenada, es decir, métodos públicos cuya seguridad dependiera en la clave y solo la persona autorizada pudiera acceder al contenido cifrado. Ante la demanda creciente por este tipo de estándares, la Oficina Nacional de Estándares de Estados Unidos (NBS) lanzó un proyecto para generar un algoritmo estándar, seguro y público. Inicialmente no recibieron muchas propuestas aceptables, hasta que IBM presentó su método denominado Lucifer que pasó los exámenes de la NBS. Después de una aceptación adicional por parte de la Agencia Nacional de Seguridad y el estudio y validación por parte de la comunidad criptográfica, el cifrado Lucifer de IBM se convirtió en un estándar de cifrado cambiando su nombre a DES, Data Encryption Estandar.

El DES era vulnerable a ataques de fuerza bruta, es decir, al probar todas las posibles claves una por una. Sin embargo, en los años sesenta, la capacidad computacional de los ordenadores era insuficiente para realizar dicho ataque en un tiempo razonable. A finales de los años 90, llegó el momento en el que el DES podía ser descifrado con un esfuerzo computacional enorme pero posible para las máquinas de la época. Este hecho llevó a la NBS a realizar una nueva búsqueda de un estándar general que sustituyera a DES. Ante esta nueva necesidad se optó por un algoritmo que había surgido anteriormente: la criptografía asimétrica o de clave pública, denominada de esta forma por funcionar con dos claves, una pública y otra secreta, además de utilizar una clave distinta para cifrar y descifrar. Cuando una persona A deseaba enviar un mensaje a otra persona B, tomaba la clave pública de B y cifraba el mensaje con dicha clave. El cifrado era completamente seguro ya que solo B era capaz de descifrar el mensaje al poseer la clave privada asociada a dicha clave pública.

Esta idea, ampliamente relacionada con las matemáticas, fue un hito en el mundo de la criptografía, provocando el auge de numerosos métodos a lo largo de los años setenta. El algoritmo más popular es el RSA, cuyas siglas provienen de sus creadores Rivest, Shamir y Adleman, pertenecientes al MIT. Los tres publicaron una propuesta en 1977 pero la NSA decidió paralizar su desarrollo iniciando acciones legales contra su publicación. Tras unos años de incertidumbre, en 1982 decidieron abrir su propia empresa con la que tuvieron un gran éxito, RSA Data Security. A día de hoy, este tipo de criptografía se encuentra en la mayoría de protocolos y algoritmos que protegen nuestras comunicaciones.

Capítulo 5

Codificación

Tras una revisión teórica de los métodos criptográficos más importantes desde la Antigüedad hasta la llegada de los primeros ordenadores, se ha llevado a cabo la codificación de algunos de los métodos con el objetivo de permitir al usuario la encriptación y desencriptación de mensajes de forma rápida. La codificación se ha realizado con el respaldo y la influencia del libro "Criptografía sin secretos con Python" de David Arboleda Brihuega [5]. Este texto aborda algunos de los sistemas contemplados en este trabajo, y su contenido ha sido fundamental para el entendimiento y la implementación de otros métodos. La codificación se ha llevado a cabo en el lenguaje de programación Python, una elección poderosa debido a su simplicidad, legibilidad y riqueza del ecosistema de sus bibliotecas.

5.1. Escítala o cifrado de transposición

El primer método criptográfico codificado ha sido el de la escítala lacedemonia (transposición simple), cuya explicación teórica puede encontrarse en la sección 3.2. De manera previa a presentar la codificación del cifrado, es importante destacar una función de la que se hará uso en la gran mayoría de métodos realizados en este trabajo. Se ha creado una función *eliminar_espacios* cuyo objetivo es eliminar los caracteres de espacio en el texto claro y aumentar la seguridad:

```
def eliminar_espacios(texto_claro):  
    nuevo_texto=''  
    for caracter in texto_claro:  
        if caracter != ' ':  
            nuevo_texto += caracter  
    return nuevo_texto
```

En la función de cifrado de la escítala o de transposición, que recibe como argumentos el texto y la clave o número de letras por fila, primero se eliminan los espacios del texto original para fortalecer la seguridad y trabajar únicamente con los caracteres que se desean cifrar. A continuación, se inicializa una lista llamada criptograma con una longitud igual al número de columnas (clave). Los

Capítulo 5. Codificación

caracteres se introducen en la columna correspondiente sumando el valor de la clave a cada posición, lo que resulta en una lista de cadenas de caracteres (*Strings*), o dicho de otra manera, una lista de secuencias de caracteres. Finalmente, la función *join* se utiliza para transformar la lista de secuencias de caracteres en una única cadena (*String*) sin separaciones entre los caracteres, y este resultado se imprime en mayúsculas.

```
def cifrar_trans(texto_claro, clave):
    #Cada fila del criptograma es una columna
    texto_claro = eliminar_espacios(texto_claro)
    criptograma = [''] * clave
    #Recorrer cada columna de la tabla
    for col in range(clave):
        pos = col
        #En cada columna se anaden las letras
        #correspondientes a los múltiplos de la clave
        while pos < len(texto_claro):
            criptograma[col] += texto_claro[pos]
            #desplazar la posición
            pos += clave
        #Convertir la lista en una única cadena
    criptograma = ''.join(criptograma)
    #Imprimir salida
    print(criptograma.upper())
```

Para el descifrado, se ha codificado otra función, llamada *descifrado_trans*, que recibe como argumentos el criptograma y la clave, la cual representa el número de letras que entran por fila. Después de eliminar los espacios, se calcula el número de columnas mediante la función *math.ceil* ya que

$$\text{columnas} = \left\lceil \frac{\text{longitud criptograma}}{\text{número de filas}} \right\rceil.$$

Después, se calculan el número de celdas vacías sabiendo que

$$\text{celdas vacías} = \text{celdas totales} - \text{longitud criptograma}.$$

Una vez se tienen ambos valores, se vuelve a inicializar la lista de cadenas con el número de columnas calculado. Se utiliza un bucle para añadir un elemento de cada cadena en lugar de añadir cada cadena por completo, como se hacía en el método de cifrado. Finalmente, la función *join* se utiliza para transformar la lista de secuencias de caracteres en una única cadena (*String*) sin separaciones entre los caracteres, y este resultado se imprime en mayúsculas.

```
def descifrar_trans(criptograma, clave):
    criptograma = eliminar_espacios(criptograma)
    #Número de columnas de la matriz
    num_col = math.ceil(len(criptograma)/clave)
    #Número de filas
```

```
num_filas = clave
#Número de celdas vacías
celdas_vacias = (num_col * num_filas) - len(criptograma)
#Cada cadena de texto es una columna
texto_claro = [''] * num_col
col = 0
fila = 0
for caracter in criptograma:
    texto_claro[col] += caracter
    col += 1 #Avanzar columna
    #Si no hay más columnas o se ha llegado a una celda
    #vacía, se vuelve a la columna inicial
    if (col == num_col) or
        (col == num_col - 1 and fila >= num_filas - celdas_vacias):
        col = 0
        fila += 1
texto_claro = ''.join(texto_claro)
print(texto_claro.upper())
```

Un ejemplo de uso de ambos métodos con el ejemplo mostrado en la sección 3.2 es el siguiente:

```
Introduce el texto: La felicidad esta en el saber
Introduce la clave: 5
LIDEAACENBFISEEEDTLRLAAS

Introduce el texto a descifrar: LIDEAACENBFISEEEDTLRLAAS
Introduce la clave: 5
LAFELICIDADESTAENELSABER
```

5.2. Cifrado Atbash

El segundo método codificado ha sido el cifrado Atbash, cuya teoría se encuentra en la sección 3.3 de este trabajo. Se debe destacar que se ha creado una variable global que incluye los 26 caracteres del alfabeto del trabajo, `ALFABETO = 'ABCDEFGHIJKLMNOPQRSTUVWXYZ'`, de la que se hará uso en la gran mayoría de cifrados de sustitución.

Para el cifrado, se ha programado una función `cifrado_atbash` que recibe como parámetro el texto claro a cifrar. Primero, se introduce una variable que representa el alfabeto de sustitución, `CIFRADO = ZYXWVUTSRQPONMLKJIHGFEDCBA`. Luego, se eliminan los espacios del texto claro, se convierte en mayúsculas y, para cada carácter, cuyo índice se obtiene en la variable global `ALFABETO`, se realiza una sustitución por el carácter con el mismo índice en la variable `CIFRADO`. Finalmente, se imprime la salida del texto cifrado.

```
def cifrado_atbash(texto_claro):
    #Alfabeto de cifrado y descifrado
```

```
CIFRADO = 'ZYXWVUTSRQPONMLKJIHGFEDCBA_'
salida = ''
texto_claro = eliminar_espacios(texto_claro)
for simbolo in texto_claro.upper():
    if simbolo in ALFABETO:
        indice = ALFABETO.index(simbolo)
        salida += CIFRADO[indice]
print(salida)
```

Una de las principales características de esta función es que es recíproca, es decir, es útil tanto para cifrar como para descifrar. La posición de los caracteres del texto cifrado en la variable `ALFABETO` se sitúan en el mismo índice que los caracteres originales en `CIFRADO`. Un ejemplo de uso con el texto plano cifrado en la sección 3.3 es:

```
Introduce un texto: La felicidad esta en el saber
OZUVORXRWZWVHGZVMVOHZYVI
```

Y su posterior descifrado:

```
Introduce un texto: OZUVORXRWZWVHGZVMVOHZYVI
LAFELICIDADESTAENELSABER
```

5.3. Cifrado de Polibio

El siguiente método codificado ha sido el cifrado de Polibio, cuya teoría puede ser consultada en la sección 3.5. En primer lugar, se ha creado la función auxiliar `generar_cuadrícula_clave`. En dicha función, inicialmente, se convierte la clave a mayúsculas y se reemplazan los caracteres J por el carácter I. Luego, los caracteres únicos de la clave, ordenados por orden de aparición, se colocan en la variable `clave`. Esto se realiza mediante la expresión: `clave = ''.join(sorted(set(clave.upper()), key=clave.upper().index))`. Posteriormente, se añaden los caracteres restantes del alfabeto y se dividen en filas de 5 elementos para representar la tabla.

```
def generar_cuadrícula_clave(clave):
    clave = clave.upper().replace('J', 'I')
    clave = ''.join(sorted(set(clave.upper()),
                           key=clave.upper().index))
    #Crear una lista de letras del alfabeto que no están en la clave
    letras_restantes = [letra for letra in ALFABETO
                        if letra != 'J' and letra not in clave]
    #Añadir letras del alfabeto a la clave
    clave_con_letras_restantes = clave + ''.join(letras_restantes)
    #Dividir la clave en filas de 5 elementos
    cuadrícula = [clave_con_letras_restantes[i:i + 5]
                  for i in range(0, len(clave_con_letras_restantes), 5)]
    return cuadrícula
```

Es importante destacar que si se desea generar una cuadrícula sin clave, la función recibirá como argumento el carácter vacío ("").

La función `cifrado_polibio` recibe como argumentos el texto claro y la clave. El primer paso es eliminar los espacios y verificar que todos los caracteres de la clave y del texto sean letras (si existe clave), ya que el cifrado de Polibio trabaja únicamente con 25 letras (fusionando la I y la J del alfabeto). En segundo lugar, se genera la cuadrícula llamando a la función `generar_cuadrícula_clave` con la clave como argumento. Luego, se crea un diccionario donde cada letra de la cuadrícula se asocia con los dos dígitos que representan su posición en la misma. Una vez construido el diccionario, se transforma el texto a cifrar en mayúsculas, se reemplazan los caracteres J por los caracteres I, y finalmente se lleva a cabo la sustitución carácter por carácter.

```
def cifrado_polibio(texto, clave):
    texto = eliminar_espacios(texto)
    #Comprobaciones previas
    if not texto.isalpha():
        print("El_texto_es_incorrecto_para_el_cifrado_de_Polibio")
        return
    if not clave.isalpha() and clave != '':
        print("La_clave_es_incorrecta_para_el_cifrado_de_Polibio")
        return
    cuadrícula = generar_cuadrícula_clave(clave)
    cuadrícula_cifrada = {}
    #Convertir la cuadrícula en un diccionario
    for i in range(len(cuadrícula)):
        for j in range(len(cuadrícula[i])):
            cuadrícula_cifrada[cuadrícula[i][j]] = f"{i+1}{j+1}"
    texto_cifrado = ""
    #Convertir las J en I
    texto = texto.upper().replace('J', 'I')
    # Cifrar el texto
    for caracter in texto:
        if caracter in cuadrícula_cifrada:
            texto_cifrado += cuadrícula_cifrada[caracter]
    print (texto_cifrado)
```

La función `descifrado_polibio` también recibe el texto a descifrar y la clave como argumento. En primer lugar, se verifica que el texto solo contenga caracteres numéricos y que su longitud sea par. Además, se verifica que la clave solo contenga caracteres alfabéticos en el caso en el que se utilice. A continuación, se utiliza la función `generar_cuadrícula_clave` para generar la cuadrícula. Luego, mediante un bucle `while`, se toman los dígitos de dos en dos para obtener las letras que se encuentran en dichas posiciones de la cuadrícula y así realizar el descifrado.

```
def descifrado_polibio(texto, clave):
    #Comprobaciones previas
```

```
if not texto.isdigit() or len(texto)%2 != 0:
    print("No_se_ha_utilizado_el_cifrado_de_Polibio")
    return
if not clave.isalpha() and clave!='':
    print("No_se_ha_utilizado_el_cifrado_de_Polibio")
    return
cuadrícula = generar_cuadrícula_clave(clave)
texto_claro = ""
i=0
#Bucle de dos en dos hasta el penúltimo carácter
while(i<len(texto)-1):
    #Se escoge la letra en dicha posición
    texto_claro += cuadrícula[int(texto[i])-1][int(texto[i+1])-1]
    i = i+2
print(texto_claro)
```

Un ejemplo de uso con el texto plano cifrado en la sección 3.5 es:

```
Introduce un texto: La felicidad esta en el saber
Introduce una clave (Enter para cifrar sin clave):
311121153124132414111415434411153315314311121542
```

Otro ejemplo para el descifrado es el siguiente:

```
Introduce un texto: 311121153124132414111415434411153315314311121542
Introduce una clave (Enter para cifrar sin clave):
LAFELICIDADESTAENELSABER
```

Y un último ejemplo de cifrado y descifrado haciendo uso de la clave "saber":

```
Introduce un texto: La felicidad esta en el saber
Introduce una clave (Enter para cifrar sin clave): saber
331223143331213122122214114412143514331112131415

Introduce un texto: 331223143331213122122214114412143514331112131415
Introduce una clave (Enter para cifrar sin clave): saber
LAFELICIDADESTAENELSABER
```

5.4. Cifrado Cesar

El cifrado César, cuya explicación teórica puede encontrarse en la sección 3.6, se ha codificado de forma sencilla en una sola función para el cifrado y descifrado, utilizando la variable *ALFABETO* presentada anteriormente. La función recibe como argumentos el *texto* a cifrar o descifrar, el *desplazamiento* o clave y el *modo* (que puede ser CIFRAR o DESCIFRAR). En cada uno de los casos se busca el carácter del texto en la variable *ALFABETO* para obtener su posición. Si el carácter no es alfabético, se incluye sin cifrar en el texto cifrado. Una vez obtenida la posición, si el modo es de cifrado, se suma el desplazamiento en módulo 26; en caso contrario, se resta.

```
def cifrado_cesar(texto, desplazamiento, modo):
    salida = ''
    texto = eliminar_espacios(texto)
    for caracter in texto.upper():
        #Comprobar si caracter está en alfabeto
        if caracter in ALFABETO:
            pos = ALFABETO.find(caracter)
            if modo == 'CIFRAR':
                pos = (pos+desplazamiento) % 26
            elif modo == 'DESCIFRAR':
                pos = (pos-desplazamiento) % 26
            salida += ALFABETO[pos]
        else:
            #Si el caracter no es una letra no se cifra
            salida += caracter
    print(salida)
```

El ejemplo de la sección 3.6, haciendo uso de la clave original que utilizaba el emperador romano Julio César, es el siguiente:

```
Introduce un texto: La felicidad esta en el saber
Introduce un desplazamiento: 3
Introduce un modo (CIFRAR o DESCIFRAR): CIFRAR
ODIHOLFLGDGHVWDHQHOVDEHU
```

```
Introduce un texto: ODIHOLFLGDGHVWDHQHOVDEHU
Introduce un desplazamiento: 3
Introduce un modo (CIFRAR o DESCIFRAR): DESCIFRAR
LAFELICIDADESTAENELSABER
```

También se ha creado la función `fuerza_bruta_cesar` para facilitar al usuario realizar un ataque por fuerza bruta con todas las posibles combinaciones (tratado en la sección 3.7). La función recibe el texto que se quiere atacar y va restando todos los posibles desplazamientos a cada uno de los caracteres.

```
def fuerza_bruta_cesar(texto):
    texto = texto.upper()
    #Probar todos los desplazamientos
    for clave in range(1, len(ALFABETO)):
        salida = ''
        for caracter in texto:
            if caracter in ALFABETO:
                pos=ALFABETO.find(caracter)
                pos = (pos-clave) % len(ALFABETO)
                salida += ALFABETO[pos]
            else:
                salida += caracter
        print('Clave_%d:_%s' % (clave, salida))
```

El ejemplo mostrado en la sección 3.7 es el siguiente:

Introduce el texto a atacar: oleur

Clave 1: NKDTQ

Clave 2: MJCSP

Clave 3: LIBRO

Clave 4: KHAQN

Clave 5: JGZPM

Clave 6: IFYOL

Clave 7: HEXNK

Clave 8: GDWMJ

Clave 9: FCVLI

Clave 10: EBUKH

Clave 11: DATJG

Clave 12: CZSIF

Clave 13: BYRHE

Clave 14: AXQGD

Clave 15: ZWPFC

Clave 16: YVOEB

Clave 17: XUNDA

Clave 18: WTM CZ

Clave 19: VSLBY

Clave 20: URKAX

Clave 21: TQJZW

Clave 22: SPIYV

Clave 23: ROHXU

Clave 24: QNGWT

Clave 25: PMFVS

5.5. Análisis de frecuencia

Dentro del criptoanálisis árabe (teoría localizada en la sección 3.7 de este trabajo) se trató el análisis de frecuencia por el cuál era posible romper un cifrado de sustitución monoalfabético. Para llevar a cabo la codificación de este método en Python lo primero que se ha hecho ha sido crear un String con las letras del alfabeto ordenadas según la frecuencia de cada letra en el idioma español, reflejadas en la figura 3.9.

```
orden_frecuencias = 'EAOLSNDRUITCPMYQBHGFVJZXWK'
```

Tras haber creado una función sencilla que cuenta el número de letras y calcula su frecuencia de aparición en el texto cifrado, llega el momento de ordenarlas. Inicialmente se crea un diccionario con las frecuencias y el número de letras que poseen esa frecuencia en el texto cifrado para proceder a su ordenación. En el caso en el que haya dos letras con frecuencias repetidas, se ordenan en orden inverso a la aparición en *orden_frecuencias* para asegurar que mensajes con la misma distribución de frecuencias produzcan idénticas salidas. Finalmente, se crea la lista que solo contiene las letras ordenadas.

```
def ordenar_frecuencias(mensaje):  
    #Sacar número de letras y frecuencia  
    frec_letras = contar_letras(mensaje)  
  
    #Construir diccionario con letras y frecuencia  
    frec_a_letras = {}  
    for letras in ALFABETO:  
        if frec_letras[letras] not in frec_a_letras:  
            frec_a_letras[frec_letras[letras]] = [letras]  
        else:  
            frec_a_letras[frec_letras[letras]].append(letras)  
    #Orden inverso al de frecuencias del español en frecuencias  
    for freq in frec_a_letras:  
        frec_a_letras[freq].sort(key=orden_frecuencias.find, reverse=True)  
        frec_a_letras[freq] = ''.join(frec_a_letras[freq])  
    #Se convierte el diccionario en lista de tuplas  
    #Se ordena por frecuencia  
    pares_frec = list(frec_a_letras.items())  
    pares_frec.sort(key=item_indice_0, reverse=True)  
    #Se crea la nueva lista con el orden del texto cifrado  
    orden_frec = []  
    for freqPair in pares_frec:  
        orden_frec.append(freqPair[1])  
    return ''.join(orden_frec)
```

El paso final es hacer uso de una función que sustituya las letras de la variable `orden_frecuencias` por las de la nueva lista que acabamos de ordenar.

```
def sustituye_frecuencia(texto_cifrado):  
    Sustitucion = ordenar_frecuencias(texto_cifrado)  
    salida = ''  
    for letra in texto_cifrado.upper():  
        if letra in ALFABETO:  
            indice = Sustitucion.index(letra)  
            salida += orden_frecuencias[indice]  
        else:  
            salida += letra  
    print(salida)
```

Cabe destacar que este método es útil para textos largos con una gran cantidad de letras y repeticiones de las mismas. En el caso en el que el texto a cifrar sea corto, será una tarea muy difícil obtener el texto claro haciendo uso de este método.

5.6. Cifrado Vigenère

Dentro de los métodos de sustitución polialfabética, se ha optado por implementar el cifrado Vigenère al representar la evolución de los métodos anteriores

de Tritemio o Bellaso. El cifrado, cuyos fundamentos teóricos se detallan en la sección 3.15 de este documento, se ha diseñado para una implementación generalizada. Esto implica que el usuario tiene la opción de proporcionar una clave de tabla personalizada, permitiéndole generar su propia tabla en lugar de depender exclusivamente de la tabla de Tritemio (véase la figura 3.13). Se ha desarrollado la función `generar_tabla` para facilitar la creación de una tabla personalizada a partir de una clave proporcionada por el usuario, y se ha creado una variable global `alfabeto = ['']*26` para inicializar dicha tabla como una lista de secuencias de caracteres. Primero, se emplean las mismas funciones que se presentan en la función `generar_cuadrícula_clave` mostrada en la sección 5.3 para eliminar los caracteres duplicados de la clave y crear un alfabeto personalizado. Luego, se procede a desplazar el alfabeto inicial una posición en cada iteración, generando así los 26 alfabetos que conforman la tabla a utilizar.

```
def generar_tabla(clave):
    #Eliminar duplicados y obtener alfabeto inicial
    clave = ''.join(sorted(set(clave.upper()),
                              key=clave.upper().index))
    letras_restantes = [letra for letra in ALFABETO
                        if letra not in clave]
    clave = clave + ''.join(letras_restantes)
    alfabeto[0] = clave
    #Rellenar el resto de alfabetos
    for i in range(1, 26):
        cadena = ''
        for j in range(0, 26):
            pos = (j + 1) % 26
            cadena += alfabeto[i - 1][pos]
        alfabeto[i] = cadena
    return alfabeto
```

Con el objetivo de comprobar el correcto funcionamiento de la función, se ha impreso una tabla que utiliza como clave la palabra clave:

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

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

Cabe destacar que si se llama a la función con una clave vacía (el carácter ""), se genera la tabla de Tritemio original (3.13).

Una vez construida la tabla, el siguiente paso es cifrar o descifrar el mensaje. Para ello, se ha desarrollado una única función que recibe como argumentos el texto a cifrar o descifrar, la clave del mensaje y el modo deseado (cifrado o descifrado). Después de convertir la clave y el texto en mayúsculas y eliminar los espacios, se procede a cifrar o descifrar el texto. Durante este proceso, se recorren los caracteres del texto y de la clave, añadiendo al criptograma la letra correspondiente a la posición indicada por la fila del carácter de la clave y la columna del carácter del texto en el caso del cifrado. En el descifrado, se realiza el proceso inverso. Si el carácter no pertenece al alfabeto utilizado en este trabajo (las 26 letras de la A a la Z, sin incluir la letra Ñ), dicho carácter no se cifra. Finalmente, se imprime el texto cifrado o descifrado.

```
def cifrado_Vigenere(clave, texto, modo):
    clave = clave.upper()
    clave = ''.join(clave.split())
    criptograma = ''
    indice_clave = 0
    texto = eliminar_espacios(texto.upper())
    for caracter in texto:
        pos = ALFABETO.find(caracter)
        #Si el caracter es una letra y está en el alfabeto
        if pos != -1:
            n = ALFABETO.find(clave[indice_clave])
            if modo == 'CIFRAR':
                criptograma += alfabeto[n][pos]
            else:
                pos = alfabeto[n].find(caracter)
                criptograma += ALFABETO[pos]
            indice_clave += 1
            if indice_clave == len(clave):
                indice_clave = 0
        else:
            criptograma += caracter
```

```
print(criptograma)
```

El ejemplo mostrado en la sección 3.15 es el siguiente:

```
Introduce el texto a cifrar: La felicidad esta en el saber
Introduce la clave de la tabla (Enter para cifrar con tabla de Tritemio):
Introduce la clave del mensaje: clave
NLFZPKNIYEFPSOEGYEGWCMEM

Introduce el texto a descifrar: NLFZPKNIYEFPSOEGYEGWCMEM
Introduce la clave de la tabla (Enter para cifrar con tabla de Tritemio):
Introduce la clave del mensaje: clave
LAFELICIDADESTAENELSABER
```

5.7. Cifrado Playfair

El cifrado Playfair, diseñado por Sir Charles Wheatstone y cuya explicación teórica puede encontrarse en la sección 3.19, ha sido implementado de una forma distinta a otros métodos que utilizan tablas. En este caso, la tabla se tratará como una lista en lugar de como una matriz, ya que es más cómodo obtener la posición exacta con *index* sin hacer uso de bucles adicionales. Para ello, en la función *cifrado_playfair*, lo primero que se hace es reemplazar las letras "I" por "J" en el texto y generar la cuadrícula con la clave, haciendo uso de la función *generar_cuadrícula_clave* utilizada anteriormente. Una vez obtenida esta cuadrícula, se aplanan para convertir el alfabeto con el que se va a trabajar en una lista. Si el texto tiene un número impar de caracteres, se añade una letra (en este caso, "W") para poder formar digramas, y se recorre el texto de dos en dos llamando a la función *digrama*.

```
def cifrado_playfair(clave, texto, modo):
    #Reemplazar J por I
    texto = texto.upper().replace('J','I')
    texto = re.sub('[^A-Z]', '', texto)
    alfabeto = generar_cuadrícula_clave(clave)
    #Convertir el alfabeto en una lista
    alfabeto = [caracter for fila in alfabeto for caracter in fila]
    print(alfabeto)
    #Añadir X al final si la longitud es impar para digramas
    if len(texto) % 2 == 1:
        texto += 'W'
    salida = ''
    #Recorrer cada dos letras
    for letra in range(0, len(texto), 2):
        di = digrama(texto[letra], texto[letra+1], alfabeto, modo)
        if di == "Existen_dos_letras_iguales_en_un_digrama":
            salida = "Existen_dos_letras_iguales_en_un_digrama"
            break
    else:
```

```

        salida += di
    print (salida)

```

Esto provoca que la tabla 5.1 se transforme en la lista 5.2.

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

Figura 5.1: Tabla con clave 'Saber'

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

Figura 5.2: Lista a partir de la tabla 5.1

La letra "S", que en la tabla ocupa la posición [0,0], ahora ocupa la posición 0. Otro ejemplo es la letra "L", que en la tabla ocupa la posición [2,3], ahora ocupa la posición 13. Por tanto, tras localizar el índice en la lista, es sencillo hallar la posición en la tabla para realizar las operaciones a partir de la división entera de la posición en la lista entre 5 para obtener la fila y la posición en la lista módulo 5 para obtener la columna. En el caso de la letra "L" se tiene que $13 // 5 = 2$ y $13 \bmod 5 \equiv 3$. Por otro lado, también es posible obtener la posición de la lista a partir de la cuadrícula haciendo $\text{fila} \cdot 5 + \text{columna}$, en el caso de la "L" sería $2 \cdot 5 + 3 = 13$.

A partir de la lógica anterior se ha implementado la función *digrama*, encargada de devolver la sustitución de cada digrama a partir de la tabla playfair que recibe como argumento en el parámetro *alfabeto*. Una vez halladas las filas y columnas de las dos letras, se inicializa un atributo posición con valor 1 si es cifrar para representar un movimiento a la derecha en los casos 1 y 2, y un valor -1 si es descifrar para representar un movimiento a la izquierda en los casos 1 y 2. Tras comprobar el caso que es, se aplican las operaciones correspondientes y se devuelve el digrama cifrado.

```

def digrama(primer_letra, segunda_letra, alfabeto, modo):

    if modo == 'cifrar':
        #No puede haber un digrama con las mismas letras
        if primera_letra==segunda_letra:
            segunda_letra='X'
    else:
        #Caso error

```

```
if primera_letra == segunda_letra:
    print("Existen_dos_letras_iguales_en_un_digrama")
    return
fila_primera, col_primera = alfabeto.index(primer_letra)//5,
                             alfabeto.index(primer_letra) %5
fila_segunda, col_segunda = alfabeto.index(segunda_letra)//5,
                             alfabeto.index(segunda_letra) %5
#Si se cifra pos = 1 (hacia derecha)
if modo=='cifrar':
    pos = 1
#Si no se cifra pos = -1 (hacia izquierda)
else:
    pos=-1

#Caso coinciden filas
if fila_primera == fila_segunda:
    return alfabeto[fila_primera*5+(col_primera+pos) %5] +
           alfabeto[fila_segunda*5+(col_segunda+pos) %5]
#Caso coinciden columnas
elif col_primera == col_segunda:
    return alfabeto[((fila_primera+pos) %5)*5+col_primera] +
           alfabeto[((fila_segunda+pos) %5)*5+col_segunda]
#Caso filas y columnas distintas
else:
    return alfabeto[fila_primera*5 + col_segunda] +
           alfabeto[fila_segunda*5+col_primera]
```

El caso de ejemplo de la sección 3.19 se prueba a continuación:

```
Introduce el texto a cifrar: La felicidad esta en el saber
Introduce la clave del mensaje: sabiduria
HIUFQEEBSBIFDNIRQUGIBIFC
```

```
Introduce el texto a descifrar: HIUFQEEBSBIFDNIRQUGIBIFC
Introduce la clave del mensaje: sabiduria
LAFELICIDADESTAENELSABER
```

5.8. Código Morse

A pesar de que el código Morse, presentado en la sección 3.21, no es muy útil para ocultar información, se ha seleccionado como un ejemplo de código para su implementación. En primer lugar, se ha creado un diccionario con cada carácter y su equivalencia en código morse llamado *morse*. Una vez creado el diccionario, se han creado dos funciones, *codificar_letra_morse* y *decodificar_letra_morse*, para codificar y decodificar los caracteres en morse. A partir de estas funciones se van recorriendo los caracteres en el texto introducido por el usuario para terminar imprimiendo el mensaje codificado o decodificado.

```
def decodificar_morse(texto):
    salida = ""
    for caracter in texto.split("_"):
        caracter_claro = decodificar_letra_morse(caracter)
        salida += caracter_claro
    print(salida)

def codificar_morse(texto_claro):
    salida = ""
    texto_claro = texto_claro.upper()
    for caracter in texto_claro:
        caracter_morse = codificar_letra_morse(caracter)
        salida += caracter_morse + "_"
    print(salida)
```

Un ejemplo con la palabra "hola" es el siguiente:

```
Introduce el texto a codificar: hola
.... --- .-.. .-
```

```
Introduce el texto a decodificar: .... --- .-.. .-
HOLA
```

5.9. Cifrado ADFGX y ADFGVX

5.9.1. ADFGX

El cifrado ADFGX, cuya historia y explicación se encuentra en la sección 3.24, se ha implementado haciendo uso de la función `generar_cuadrícula_clave` utilizada para el cifrado de Polibio en la sección 5.3. Este método es el encargado de crear la tabla con 25 elementos (todas las letras del alfabeto juntando la I y la J en la misma casilla) utilizada para la primera parte del cifrado. Además de esto, se ha creado una función `obtener_ADFGX_sustitucion` que recibe como argumento la cuadrícula y devuelve un diccionario con cada letra y sus respectivas coordenadas en ADFGX.

```
def obtener_ADFGX_sustitucion(cuadrícula):
    diccionario = {}
    coordenadas = 'ADFGX'
    for i, fila in enumerate(cuadrícula):
        for j, letra in enumerate(fila):
            diccionario[letra] = coordenadas[i] + coordenadas[j]
    return diccionario
```

Haciendo uso de estas dos funciones, es posible obtener el texto sustituido. A continuación, se procede con la parte de la transposición. Primero, se halla el número de columnas (la longitud de la clave) y el número de filas (calculado como $(\text{longitud del mensaje} + \text{columnas} - 1) // \text{columnas}$). Luego, se crea una

Capítulo 5. Codificación

lista de listas utilizando la instrucción `mensaje_matriz = [''] * columnas`, en la que se van incluyendo las letras del mensaje. Finalmente, se reordenan las columnas en función del orden de las letras de la clave y se concatenan las columnas de forma ordenada para obtener el mensaje cifrado, gracias a la línea `mensaje_cifrado = ''.join(mensaje_matriz[i] for _, i in clave_ordenada)`. La función `cifrado_adfgx`, cuyos parámetros son el mensaje a cifrar, la clave de la tabla y la clave de la transposición, es la encargada de llevar a cabo todo este proceso.

```
def cifrado_adfgx(mensaje, clave_cuadrícula, clave_transposición):
    mensaje = mensaje.upper().replace('J', 'I')
    cuadrícula = generar_cuadrícula_clave(clave_cuadrícula)
    diccionario_sustitucion = obtener_ADFGX_sustitucion(cuadrícula)
    #Sustitución
    mensaje_sustituido = ''.join(
        diccionario_sustitucion[letra] for letra in mensaje
        if letra in diccionario_sustitucion)

    #Transposición
    columnas = len(clave_transposición)
    #Filas para asegurar que caben todas las letras
    filas = (len(mensaje_sustituido)+columnas-1) // columnas
    mensaje_matriz = [''] * columnas
    for i, letra in enumerate(mensaje_sustituido):
        columna = i % columnas
        mensaje_matriz[columna] += letra
    #Ordenar columnas según la clave de transposición
    clave_ordenada = sorted((e, i) for i, e
        in enumerate(clave_transposición))
    mensaje_cifrado = ''.join(mensaje_matriz[i] for _,
        i in clave_ordenada)

    print(mensaje_cifrado)
    return mensaje_cifrado
```

Para el descifrado, es necesario realizar el proceso inverso. Primero, se halla el inverso del diccionario mediante `{v: k for k, v in diccionario.items()}` después de haber obtenido el diccionario original de cifrado. Luego, se debe rellenar la tabla por columnas de forma ordenada. Para esto, es necesario calcular la longitud de cada columna, determinando el número de filas $(\text{longitud del cifrado} + \text{columnas} - 1) // \text{columnas}$ y las letras sobrantes $(\text{longitud del texto cifrado} \% \text{columnas})$.

También se debe determinar el orden de inserción a partir de la clave utilizando `sorted((caracter, i) for i, caracter in enumerate(clave_transposición))`. Esto devuelve una lista de tuplas cuyo primer elemento es el carácter y el segundo es la posición que ocupa en la clave. Un ejemplo para la palabra clave es el siguiente:

```
[('a', 2), ('c', 0), ('e', 4), ('l', 1), ('v', 3)]
```

Tras haber rellenado las columnas según el orden determinado previamente, se lee el texto por filas y se realiza la sustitución utilizando el diccionario inverso calculado al principio. Estas operaciones se recogen en la función `descifrado_adfgx`, cuyos parámetros son el texto a descifrar, la clave de la tabla y la clave de la transposición.

```
def descifrado_adfgx(mensaje_cifrado, clave_cuadrícula, clave_transposición):
    mensaje_cifrado = mensaje_cifrado.upper()
    cuadrícula = generar_cuadrícula_clave(clave_cuadrícula)
    diccionario = obtener_ADFGX_sustitucion(cuadrícula)
    diccionario_inverso = {v: k for k, v in diccionario.items()}
    #Transposición inversa
    columnas = len(clave_transposición)
    filas = (len(mensaje_cifrado) + columnas - 1) // columnas
    longitudes_columnas = [filas] * columnas
    sobrantes = len(mensaje_cifrado) % columnas
    #Longitud de cada columna
    if sobrantes != 0:
        for i in range(columnas-1, sobrantes-1, -1):
            longitudes_columnas[i] -= 1
    orden_clave = sorted((caracter, i) for i,
                        caracter in enumerate(clave_transposición))
    mensaje_matriz = [''] * columnas
    index = 0
    #Rellenar matriz
    for _, i in orden_clave:
        mensaje_matriz[i] = mensaje_cifrado[index:index +
                                            longitudes_columnas[i]]
        index += longitudes_columnas[i]
    maximo = max(longitudes_columnas)
    #Rellenar con espacios para evitar errores
    mensaje_matriz = [col + ' ' * (maximo - len(col))
                      for col in mensaje_matriz]
    mensaje_transpuesto = ''.join(mensaje_matriz[i][j]
                                   for j in range(maximo)
                                   for i in range(len(mensaje_matriz)))
    mensaje_transpuesto = mensaje_transpuesto.replace(" ", "")
    #Sustitución inversa
    mensaje_descifrado = ''.join(diccionario_inverso
                                  [mensaje_transpuesto[i:i + 2]]
                                  for i in range(0, len(mensaje_transpuesto), 2))
    print(mensaje_descifrado)
    return mensaje_descifrado
```

El ejemplo de la sección 3.24 es el siguiente:

```
Introduzca el texto a cifrar: felicidad
Introduzca la clave de la tabla: sabiduria
Introduzca la clave de la transposición: clave
```

Capítulo 5. Codificación

DGAXDGADFFAXAGAGDX

Introduzca el texto a descifrar: DGAXDGADFFAXAGAGDX

Introduzca la clave de la tabla: sabiduria

Introduzca la clave de la transposición: clave

FELICIDAD

5.9.2. ADFGVX

Al igual que se explica en la sección 3.24, los alemanes decidieron introducir una fila y columna más en medio de la guerra para incluir los números del 0 al 9, llamando al nuevo cifrado ADFGVX. Un ejemplo de nueva tabla con la clave sabiduría es el siguiente:

	A	D	F	G	V	X
A	S	A	B	I	D	U
D	R	C	E	F	G	H
F	J	K	L	M	N	O
G	P	Q	T	V	W	X
V	Y	Z	0	1	2	3
X	4	5	6	7	8	9

Figura 5.3: Ejemplo de tabla ADFGVX con clave "sabiduría"

Para implementar este cifrado en Python se ha reutilizado la estructura de las funciones de cifrado y descifrado presentadas anteriormente. Solo se han llevado a cabo modificaciones en el uso de un nuevo alfabeto, inicializado como `ALFABETO_ADFGVX = 'ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789'`, la función de `generar_cuadricula_clave6`, que ha sido adaptada para crear una matriz 6x6 con el nuevo alfabeto, y la función `obtener_ADFGVX_sustitucion`, encargada de crear el diccionario de sustitución con el nuevo alfabeto.

```
def generar_cuadricula_clave6(clave):  
    clave = ''.join(sorted(set(clave.upper()),  
                           key=clave.upper().index))  
    #Crear una lista de letras del alfabeto que no están en la clave  
    letras_restantes = [letra for letra in ALFABETO_ADFGVX  
                        if letra not in clave]  
    #Añadir letras del alfabeto a la clave  
    clave_con_letras_restantes = clave + ''.join(letras_restantes)  
    #Dividir la clave en filas de 6 elementos  
    cuadricula = [clave_con_letras_restantes[i:i + 6] for  
                  i in range(0, len(clave_con_letras_restantes), 6)]
```

```
return cuadrícula
```

```
def obtener_ADFGVX_sustitucion(cuadrícula):  
    diccionario = {}  
    coordenadas = 'ADFGVX'  
    for i, fila in enumerate(cuadrícula):  
        for j, letra in enumerate(fila):  
            diccionario[letra] = coordenadas[i] + coordenadas[j]  
    return diccionario
```

Un ejemplo con la palabra "felicidad", la clave de tabla "sabiduría" y la clave de transposición "clave" es el siguiente:

```
Introduzca el texto a cifrar: felicidad  
Introduzca la clave de la tabla: sabiduria  
Introduzca la clave de la transposición: clave  
DGAVDFAFDAGAGAFDV
```

```
Introduzca el texto a descifrar: DGAVDFAFDAGAGAFDV  
Introduzca la clave de la tabla: sabiduria  
Introduzca la clave de la transposición: clave  
FELICIDAD
```

5.10. Máquina Enigma

Se ha llevado a cabo la implementación de dos tipos de máquinas Enigma: por un lado, la máquina Enigma de tres rotores utilizada inicialmente por el ejército alemán, y por otro lado, la máquina Enigma de cuatro rotores empleada por la Armada alemana.

5.10.1. Máquina Enigma M3

Se ha codificado la máquina Enigma de 3 rotores, cuya teoría se encuentra en la sección 4.3. Para conseguir replicar la máquina Enigma de la forma más fiel posible, primero se ha realizado una extensa investigación acerca de los rotores reales utilizado por el bando alemán. Esta investigación ha llevado a la recopilación de las conexiones de rotores, caracteres de cambio y conexiones de reflectores reales. Se han representado mediante listas o tuplas para su uso en Python.

Se ha empezado definiendo una función *vuelta* y una función *cableado_inverso*, encargadas de hallar los caminos inversos de cada uno de los cableados de los rotores. Se trata de funciones sencillas que calculan la sustitución de cada letra en el alfabeto que se está utilizando durante todo el trabajo (variable *ALFABETO*) y devuelve una tupla con el cableado inverso de los 8 rotores.

Capítulo 5. Codificación

```
def vuelta(cableado):
    #Por rotor, se calcula el camino de vuelta
    salida = ''
    for i in ALFABETO:
        #Sustitución en alfabeto
        salida += ALFABETO[cableado.find(i)]
    return salida

def cableado_inverso():
    #Tupla con el recorrido inverso de los rotores
    inverso = []
    for i in range(len(cableado_rotor)):
        inverso.append(vuelta(cableado_rotor[i]))
    inverso = tuple(inverso)
    return inverso
```

Otra de las funciones auxiliares que se utiliza es la función *numero*, encargada de devolver el número que representa una letra (del 0 al 25) a partir de la creación de un diccionario.

```
def numero(car):
    car = car.upper()
    arr = {'A':0, 'B':1, 'C':2, 'D':3, 'E':4, 'F':5, 'G':6, 'H':7, 'I':8,
           'J':9, 'K':10, 'L':11, 'M':12, 'N':13, 'O':14, 'P':15, 'Q':16,
           'R':17, 'S':18, 'T':19, 'U':20, 'V':21, 'W':22, 'X':23, 'Y':24, 'Z':25}
    return arr[car]
```

Finalmente, la función auxiliar *sustituye* es la encargada de llevar a cabo la sustitución de cada componente de la máquina Enigma sumando o restando el número de veces que haya avanzado la máquina desde la configuración inicial.

```
def sustituye(letra, alfabeto=ALFABETO, veces=0):
    #Sustituye una letra en función del número de movimientos
    indice = (numero(letra) + veces) % 26
    return alfabeto[indice]
```

Una vez definidas todas las funciones auxiliares, se procede a presentar las funciones de cada uno de los componentes de la máquina Enigma.

En primer lugar aparece la función *rotor*, cuyo objetivo es simular el comportamiento del rotor haciendo uso de la función *sustituye*. Primero calcula el carácter en el que está el rotor y lleva a cabo su sustitución en *ALFABETO*.

```
def rotor(letra, veces, cableado_rotor):
    #Letra que corresponde en el rotor
    letra = sustituye(letra, cableado_rotor, veces)
    #Letra de salida por el rotor
    return sustituye(letra, veces=-veces)
```

La función *refleja* simula el funcionamiento del reflector, sustituyendo a partir del cableado del reflector la letra resultante.

```
def refleja(letra):
    #Simula el reflector
    return sustituye(letra, cableado_reflector[reflector])
```

La función `aplicar_clavijero` se encarga de simular el clavijero de la máquina intercambiando las letras conectadas.

```
def aplicar_clavijero(letra):
    for clavija in clavijero:
        if letra == clavija[0]: return clavija[1]
        if letra == clavija[1]: return clavija[0]
    return letra
```

El avance del rotor se simula mediante la función `avanza_rotor`. Se consideran los casos en los que el rotor derecho o rápido alcanza la muesca, lo que provoca el incremento del rotor central, y en los que el rotor central alcanza la muesca, lo que provoca el incremento del rotor izquierdo.

```
def rotor_avanza():
    #Los rotores mueven al siguiente rotor dependiendo de la muesca
    if inicio[1] in muesca[rotores[1]]:
        #Incrementa rotores izquierda y central
        inicio[0] = sustituye(inicio[0], veces= 1)
        inicio[1] = sustituye(inicio[1], veces= 1)
    if inicio[2] in muesca[rotores[2]]:
        #Incrementa rotor central
        inicio[1] = sustituye(inicio[1], veces = 1)
    #Incrementa rotor derecho (siempre)
    inicio[2] = sustituye(inicio[2], veces=1)
```

La función `cifrar_caracter` se construye utilizando todas las funciones anteriores, y su objetivo principal es simular el movimiento del impulso eléctrico a través de los distintos componentes. Inicialmente, se produce el avance de los rotores en la máquina. A continuación, el impulso llega al clavijero, atraviesa los rotores, se refleja en el reflector, vuelve a pasar por los rotores y, finalmente, retorna al clavijero para mostrar la letra resultante del cifrado.

```
def cifrar_caracter(letra):
    #Con cada letra los rotores avanzan
    rotor_avanza()
    #Entrada al clavijero
    letra = aplicar_clavijero(letra)
    #Camino de ida de la señal por rotores
    for i in [2,1,0]:
        veces = ord(inicio[i]) - ord(posicion_interna [i])
        letra = rotor(letra, veces, cableado_rotor[rotores[i]])
    #Entrada y salida por el reflector
    letra = refleja(letra)
    #Camino de vuelta por rotores
```

```
for i in [0,1,2]:
    veces = ord(inicio[i]) - ord(posicion_interna[i])
    letra = rotor(letra, veces, inverso[rotores[i]])
#Salida por el clavijero
letra = aplicar_clavijero(letra)
#Letra resultante
return letra
```

Para cada carácter del texto introducido por el usuario, se aplica la función *cifrar_caracter* para obtener el resultado final. Cabe destacar que una de las principales particularidades de la máquina Enigma es su reciprocidad, es decir, que la máquina puede ser utilizada tanto para el cifrado como para el descifrado con la misma clave. Esto implica que la función codificada es igualmente válida para ambas tareas.

5.10.2. Máquina Enigma M4 o SHARK

Adicionalmente, se ha implementado la máquina Enigma de cuatro rotores, denominada Sistema SHARK o Enigma M4. El código ha sufrido ligeras modificaciones respecto al de la máquina Enigma de tres rotores, dado que el funcionamiento interno era idéntico. Además, se ha investigado sobre los nuevos rotores reales y su cableado para lograr una representación fiel y auténtica.

Las dos únicas funciones que varían son *rotor_avanza* y *cifrar_caracter*. En *rotor_avanzaM4*, se muestra el avance del rotor derecho y de los dos centrales, pero el rotor izquierdo, que solía ser el lugar ocupado por los rotores denominados gamma y beta, no rota. Por otro lado, en *cifrar_caracterM4*, solo se modifica el bucle de rotores para incluir una iteración más, avanzando de 3 a 0 en lugar de avanzar de 2 a 0.

```
def rotor_avanzaM4():
    # El rotor Beta o Gamma, inicio[0], nunca se movía durante el proceso
    if inicio[2] in muesca[rotores[2]]:
        inicio[1] = sustituye(inicio[1], veces=1)
        inicio[2] = sustituye(inicio[2], veces=1)
    if inicio[3] in muesca[rotores[3]]:
        inicio[2] = sustituye(inicio[2], veces=1)
        inicio[3] = sustituye(inicio[3], veces=1)

def cifrar_caracterM4(letra):
    #Con cada letra los rotores avanzan
    rotor_avanza()
    #Entrada al clavijero
    letra = aplicar_clavijero(letra)
    #Camino de ida de la señal por rotores
    for i in [3, 2, 1, 0]:
        veces = ord(inicio[i]) - ord(posicion_interna[i])
        letra = rotor(letra, veces, cableado_rotor[rotores[i]])
    #Entrada y salida del reflector
```

```
letra = refleja(letra)
#Camino de vuelta por rotores
for i in [0, 1, 2, 3]:
    veces = ord(inicio[i]) - ord(posicion_interna[i])
    letra = rotor(letra, veces, cableado_inverso[rotores[i]])
#Salida por el claviijero
letra = aplicar_claviijero(letra)
#Letra resultante
return letra
```

El funcionamiento para el usuario es igual que el de la máquina Enigma de 3 rotores pero solicitándole una letra y un rotor adicional.

5.11. Interfaz gráfica

A pesar del poco tiempo del que se dispone para desarrollar el Trabajo de Fin de Grado, se ha implementado una interfaz gráfica básica con el objetivo de hacer más sencillo al usuario la interacción con los distintos sistemas criptográficos sin tener que trabajar con la consola. No se tenía experiencia previa en el desarrollo de interfaces gráficas por lo que se llevó a cabo una investigación de las diferentes librerías y opciones que ofrecía Python para su desarrollo. Tras evaluar distintas opciones, la elección final fue la librería PyQt5 de Python, conocida por su flexibilidad en la creación de aplicaciones de escritorio. A través de diversos tutoriales [28] se aprendieron las funciones básicas y se comenzó con el desarrollo de la aplicación.

La pantalla inicial presenta tres botones que ofrecen la posibilidad al usuario de cifrar un texto, descifrarlo o criptoanalizarlo. Al seleccionar una de estas opciones, aparecen las distintas opciones de cifrados o sistemas que se pueden utilizar. Se ha optado por incluir para cada cifrado una pantalla que incluye una breve información del cifrado que se va a utilizar, un campo para incluir el texto a cifrar o descifrar, campos para las claves necesarias de cada sistema, y un campo de resultado de solo lectura que refleja el resultado del cifrado o descifrado. Adicionalmente, para los sistemas que hacen uso de tablas, también se ha representado la tabla que se está utilizando con el fin de hacer ver al usuario el efecto que tienen las claves de tabla sobre el cifrado. Todos los *print* de los métodos anteriores se han sustituido por *returns* para que aparezcan en la interfaz gráfica.

Para cada pantalla se han creado tres layouts, uno de ellos horizontal para incluir el título del sistema, y dos verticales, uno de ellos con la información y la tabla en el caso de haberla, y otro con los campos para realizar la operación criptográfica. Las etiquetas de los campos y la información se han implementado utilizando widgets de tipo `QLabel`. Los campos destinados al texto y al resultado se han implementado con widgets de tipo `QTextEdit`, restringiendo el segundo de ellos a solo lectura. Los campos para las claves se han diseñado con widgets de tipo `QLineEdit`, dado que las claves suelen ser mucho más cortas que el texto con el que se va a operar. Los desplegables para la máquina Enigma se

han implementado haciendo uso de widgets de tipo `QComboBox`. Los botones se han implementado utilizando widgets de tipo `QPushButton`. Todas las pantallas, excepto la inicial, disponen de un botón de volver que lleva a la página anterior, ofreciéndole al usuario control y libertad para navegar por la aplicación.

Como anteriormente no se han presentado ejemplos de la máquina Enigma, se presentan a continuación en la interfaz gráfica para tratar algunos aspectos del diseño realizado. Al acceder a la pantalla de cifrado, aparece la información relativa a la máquina Enigma, incluyendo un campo para introducir el texto en claro y menús desplegables para seleccionar los rotores, sus posiciones iniciales, la clave del mensaje y las conexiones del claviero. Además, se presenta un campo de solo lectura que muestra el resultado obtenido (Figura 5.4).

Máquina Enigma M3

Información sobre la máquina Enigma M3

La máquina Enigma, creada por Arthur Scherbius, utilizaba el concepto de los discos cifradores sobre el que había trabajado Alberti en el siglo XV. Constaba de un teclado de 26 caracteres, un claviero interno de 6 pares de conexiones, un panel luminoso con 26 caracteres, 3 rotores y un reflector. Su funcionamiento era sencillo: el usuario pulsaba una letra del teclado y se enviaba un impulso eléctrico al claviero, en el que las conexiones redirigían el impulso hasta los rotores. El impulso se propagaba de derecha a izquierda a través de los cables de cada rotor y, tras ser reflejado, regresaba de izquierda a derecha. Con cada pulsación de tecla por parte del usuario, el rotor derecho avanzaba una posición en la máquina, lo que contribuía a evitar la repetición de cifrados. Después de que el primer rotor completaba una vuelta, pasaba a rotar el rotor a su izquierda, y así sucesivamente. Este mecanismo aseguraba que los ciclos antes de volver a la posición inicial fueran considerablemente más largos. Finalmente, el impulso llegaba al claviero y se dirigía al panel luminoso iluminando la letra correspondiente al cifrado. Para cifrar un texto, introducía el texto claro, los rotores a utilizar y su orden de izquierda a derecha, la posición inicial del libro de claves, la clave del mensaje y las conexiones del claviero. El resultado del panel luminoso aparecerá en la casilla de resultado. Una de las principales características de la máquina Enigma es que era recíproca, por lo que en esta pantalla también es posible descifrar mensajes.

Texto claro:

Orden de rotores:

111

Posición inicial de los anillos:

AAA

Clave de mensaje:

AAA

Parejas de letras conectadas en el claviero:

Cifrar

Resultado:

Volver

Figura 5.4: Pantalla de la aplicación para cifrar con la máquina Enigma M3

Para poder utilizar las funciones presentadas en este capítulo, se ha hecho uso de los métodos `toPlainText()` y `currentText()`, cuyo objetivo es transformar en un `String` el texto y las selecciones del usuario respectivamente. La decisión de utilizar menús desplegables en lugar de campos de texto para los parámetros de la máquina Enigma se fundamenta en la simplicidad para el usuario, la reducción de errores y el objetivo de realizar una simulación lo más realista posible. Si el usuario solo introduce el texto claro y no modifica más parámetros, el texto se cifrará con los rotores (1,1,1), la posición de los anillos (A,A,A), la clave de mensaje (A,A,A) y ninguna conexión en el claviero.

A pesar de haber incluido desplegables en el claviero para reducir los errores, también se ha creado una pequeña función que elimina las selecciones ya marcadas en el resto de desplegables para evitar que el usuario vuelva a realizar una conexión involucrando a una letra ya conectada. La función primero recopila las selecciones, bloquea las señales de los desplegables para realizar modificacio-

nes, recorre cada una de las selecciones y evalúa si ya ha sido utilizada para no incluirla en el desplegable (salvo que sea el texto actual o el texto nulo que indica que no se ha realizado conexión alguna). Finalmente, se vuelven a habilitar las señales.

```
def actualizar_combos(self):
    #Selecciones actuales
    selecciones = [combo.currentText() for combo in self.combos]
    #Actualizar las opciones en todos los QComboBox
    for combo in self.combos:
        combo.blockSignals(True)
        current_text = combo.currentText()
        combo.clear()
        for opcion in self.opciones:
            if opcion not in selecciones or opcion == current_text
            or opcion == "":
                combo.addItem(opcion)
        combo.setCurrentText(current_text)
        combo.blockSignals(False)
```

Un ejemplo del funcionamiento de esta función se presenta en la Figura 5.5, donde se observa que, al seleccionar la letra "B" en un desplegable, esta opción no está disponible para su selección en otros desplegables.

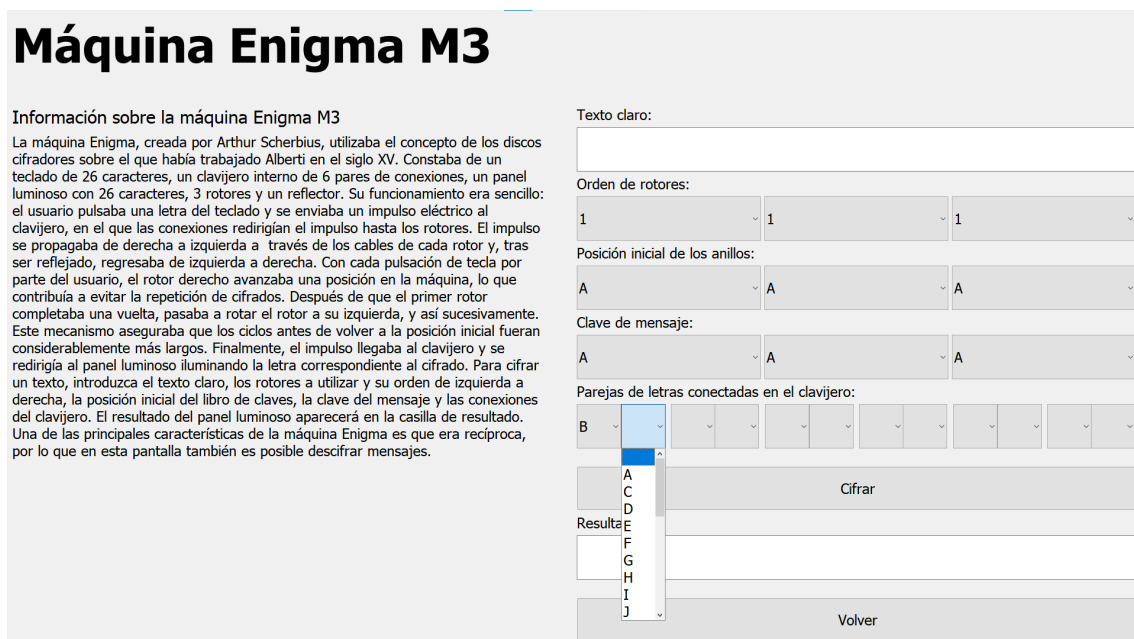


Figura 5.5: Caso de repetición en el claviero

También se ha tenido en cuenta el caso en el que el usuario introduzca una letra en uno de los desplegables del claviero y no lo haga en su pareja. En las figuras 5.6 y 5.7 se observa que, a pesar de incluir una letra en una de las parejas del claviero, el resultado no se modifica y la letra "A" no se sustituye por el carácter

vacío.

Máquina Enigma M3

Información sobre la máquina Enigma M3

La máquina Enigma, creada por Arthur Scherbius, utilizaba el concepto de los discos cifradores sobre el que había trabajado Alberti en el siglo XV. Constaba de un teclado de 26 caracteres, un clavijero interno de 6 pares de conexiones, un panel luminoso con 26 caracteres, 3 rotores y un reflector. Su funcionamiento era sencillo: el usuario pulsaba una letra del teclado y se enviaba un impulso eléctrico al clavijero, en el que las conexiones redirigían el impulso hasta los rotores. El impulso se propagaba de derecha a izquierda a través de los cables de cada rotor y, tras ser reflejado, regresaba de izquierda a derecha. Con cada pulsación de tecla por parte del usuario, el rotor derecho avanzaba una posición en la máquina, lo que contribuía a evitar la repetición de cifrados. Después de que el primer rotor completaba una vuelta, pasaba a rotar el rotor a su izquierda, y así sucesivamente. Este mecanismo aseguraba que los ciclos antes de volver a la posición inicial fueran considerablemente más largos. Finalmente, el impulso llegaba al clavijero y se redirigía al panel luminoso iluminando la letra correspondiente al cifrado. Para cifrar un texto, introducía el texto claro, los rotores a utilizar y su orden de izquierda a derecha, la posición inicial del libro de claves, la clave del mensaje y las conexiones del clavijero. El resultado del panel luminoso aparecerá en la casilla de resultado. Una de las principales características de la máquina Enigma es que era recíproca, por lo que en esta pantalla también es posible descifrar mensajes.

Texto claro:
HOLA

Orden de rotores:
1 1 1

Posición inicial de los anillos:
A A A

Clave de mensaje:
A A A

Parejas de letras conectadas en el clavijero:
A A A A A A A A

Cifrar

Resultado:
GAHG

Volver

Figura 5.6: Caso de ejemplo con valores por defecto y el texto claro "HOLA"

Máquina Enigma M3

Información sobre la máquina Enigma M3

La máquina Enigma, creada por Arthur Scherbius, utilizaba el concepto de los discos cifradores sobre el que había trabajado Alberti en el siglo XV. Constaba de un teclado de 26 caracteres, un clavijero interno de 6 pares de conexiones, un panel luminoso con 26 caracteres, 3 rotores y un reflector. Su funcionamiento era sencillo: el usuario pulsaba una letra del teclado y se enviaba un impulso eléctrico al clavijero, en el que las conexiones redirigían el impulso hasta los rotores. El impulso se propagaba de derecha a izquierda a través de los cables de cada rotor y, tras ser reflejado, regresaba de izquierda a derecha. Con cada pulsación de tecla por parte del usuario, el rotor derecho avanzaba una posición en la máquina, lo que contribuía a evitar la repetición de cifrados. Después de que el primer rotor completaba una vuelta, pasaba a rotar el rotor a su izquierda, y así sucesivamente. Este mecanismo aseguraba que los ciclos antes de volver a la posición inicial fueran considerablemente más largos. Finalmente, el impulso llegaba al clavijero y se redirigía al panel luminoso iluminando la letra correspondiente al cifrado. Para cifrar un texto, introducía el texto claro, los rotores a utilizar y su orden de izquierda a derecha, la posición inicial del libro de claves, la clave del mensaje y las conexiones del clavijero. El resultado del panel luminoso aparecerá en la casilla de resultado. Una de las principales características de la máquina Enigma es que era recíproca, por lo que en esta pantalla también es posible descifrar mensajes.

Texto claro:
HOLA

Orden de rotores:
1 1 1

Posición inicial de los anillos:
A A A

Clave de mensaje:
A A A

Parejas de letras conectadas en el clavijero:
A A A A A A A A

Cifrar

Resultado:
GAHG

Volver

Figura 5.7: Caso de ejemplo de la figura 5.6 sin variación de resultado al incluir "A" en el clavijero

Para comprobar el correcto funcionamiento de la codificación de la máquina Enigma, se presenta un caso con parámetros elegidos al azar y su posterior descifrado aprovechando la reciprocidad de la máquina Enigma en las figuras

5.8 y 5.9. Esta característica permite que desde la misma pantalla sea posible cifrar y descifrar un texto, algo mencionado en la información del cifrado.

Máquina Enigma M3

Información sobre la máquina Enigma M3

La máquina Enigma, creada por Arthur Scherbius, utilizaba el concepto de los discos cifradores sobre el que había trabajado Alberti en el siglo XV. Constaba de un teclado de 26 caracteres, un clavijero interno de 6 pares de conexiones, un panel luminoso con 26 caracteres, 3 rotores y un reflector. Su funcionamiento era sencillo: el usuario pulsaba una letra del teclado y se enviaba un impulso eléctrico al clavijero, en el que las conexiones redirigían el impulso hasta los rotores. El impulso se propagaba de derecha a izquierda a través de los cables de cada rotor y, tras ser reflejado, regresaba de izquierda a derecha. Con cada pulsación de tecla por parte del usuario, el rotor derecho avanzaba una posición en la máquina, lo que contribuía a evitar la repetición de cifrados. Después de que el primer rotor completaba una vuelta, pasaba a rotar el rotor a su izquierda, y así sucesivamente. Este mecanismo aseguraba que los ciclos antes de volver a la posición inicial fueran considerablemente más largos. Finalmente, el impulso llegaba al clavijero y se redirigía al panel luminoso iluminando la letra correspondiente al cifrado. Para cifrar un texto, introducía el texto claro, los rotores a utilizar y su orden de izquierda a derecha, la posición inicial del libro de claves, la clave del mensaje y las conexiones del clavijero. El resultado del panel luminoso aparecerá en la casilla de resultado. Una de las principales características de la máquina Enigma es que era recíproca, por lo que en esta pantalla también es posible descifrar mensajes.

Texto claro:
HOLA

Orden de rotores:
6 3 3

Posición inicial de los anillos:
D F V

Clave de mensaje:
D U C

Parejas de letras conectadas en el clavijero:
A B D W I C

Cifrar

Resultado:
YVWH

Volver

Figura 5.8: Caso de ejemplo con valores al azar y la palabra "HOLA"

Máquina Enigma M3

Información sobre la máquina Enigma M3

La máquina Enigma, creada por Arthur Scherbius, utilizaba el concepto de los discos cifradores sobre el que había trabajado Alberti en el siglo XV. Constaba de un teclado de 26 caracteres, un clavijero interno de 6 pares de conexiones, un panel luminoso con 26 caracteres, 3 rotores y un reflector. Su funcionamiento era sencillo: el usuario pulsaba una letra del teclado y se enviaba un impulso eléctrico al clavijero, en el que las conexiones redirigían el impulso hasta los rotores. El impulso se propagaba de derecha a izquierda a través de los cables de cada rotor y, tras ser reflejado, regresaba de izquierda a derecha. Con cada pulsación de tecla por parte del usuario, el rotor derecho avanzaba una posición en la máquina, lo que contribuía a evitar la repetición de cifrados. Después de que el primer rotor completaba una vuelta, pasaba a rotar el rotor a su izquierda, y así sucesivamente. Este mecanismo aseguraba que los ciclos antes de volver a la posición inicial fueran considerablemente más largos. Finalmente, el impulso llegaba al clavijero y se redirigía al panel luminoso iluminando la letra correspondiente al cifrado. Para cifrar un texto, introducía el texto claro, los rotores a utilizar y su orden de izquierda a derecha, la posición inicial del libro de claves, la clave del mensaje y las conexiones del clavijero. El resultado del panel luminoso aparecerá en la casilla de resultado. Una de las principales características de la máquina Enigma es que era recíproca, por lo que en esta pantalla también es posible descifrar mensajes.

Texto claro:
YVWH

Orden de rotores:
6 3 3

Posición inicial de los anillos:
D F V

Clave de mensaje:
D U C

Parejas de letras conectadas en el clavijero:
A B D W I C

Cifrar

Resultado:
HOLA

Volver

Figura 5.9: Caso de ejemplo de descifrado de la figura 5.8 aprovechando la reciprocidad

De la misma forma que la máquina Enigma M3, se han creado pantallas de cifrado y de descifrado con tratamiento de errores para todos los métodos presentados en este capítulo. En el capítulo 6, referente a resultados y conclusiones, se

Capítulo 5. Codificación

ofrece una visión más general y completa de la interfaz y de su funcionamiento para la selección de modo y sistema.

Capítulo 6

Resultados y conclusiones

En el trabajo se han estudiado los sistemas criptográficos previos al auge de la computación, ofreciendo una visión general de la evolución que ha sufrido la disciplina. Este estudio ha permitido una comprensión más profunda del funcionamiento de dichos sistemas y de la importancia que ha tenido cada uno de ellos a lo largo de la historia. Ha sido posible comprender la relación entre las matemáticas y la criptografía, estudiando como la relación entre las dos disciplinas se fue haciendo más palpable hasta llegar a su máximo apogeo en la Segunda Guerra Mundial, en la que los matemáticos jugaron un papel crucial. También ha permitido observar los errores que llevaron a la ruptura de métodos que a priori eran seguros, como el telegrama Zimmermann o la máquina Enigma, siendo conveniente recordar la famosa frase del filósofo español George Santayana: "Aquellos que no recuerdan el pasado están condenados a repetirlo".

Este trabajo también busca enaltecer la figura de eminencias que no han recibido el reconocimiento suficiente y dar valor a los hitos matemáticos relacionados con el criptoanálisis que impulsaron la evolución. Sin el trabajo de los criptoanalistas árabes para los cifrados de sustitución monalfabética, el estudio de Babbage y Kasiski para romper el cifrado Vigenère, el esfuerzo de los criptoanalistas polacos para atacar la primera versión de la máquina Enigma, la ruptura definitiva de la Enigma por parte de Alan Turing o la habilidad de Tutte para descifrar la máquina Lorenz, la disciplina se hubiera quedado estancada. La comprensión de estos sucesos permite apreciar la importancia de la carrera entre criptoanalistas y criptógrafos.

Adicionalmente a todo el estudio y la revisión realizada, se han implementado diversos sistemas en una aplicación que busca acercar al usuario a la criptografía desde un punto de vista distinto. En esta época de tecnologías y de la inmediatez, el diseño de una aplicación puede ayudar a los más jóvenes a interesarse más por el tema, probando en vivo como funciona el sistema del que acaban de leer la información en lugar de consultar libros y documentos de la época, algo que cada día es menos habitual. El trabajo extra realizado para conseguir implementar una interfaz gráfica se traduce en un alivio para usuarios menos vinculados a la informática.

Capítulo 6. Resultados y conclusiones

Al entrar, el usuario tiene la opción de elegir si quiere cifrar, descifrar o criptoanalizar un texto como se aprecia en la figura 6.1.



Figura 6.1: Pantalla de Inicio de la aplicación

Cuando el usuario elige una de las opciones, accede a una nueva pantalla con todos los sistemas disponibles para la opción que ha elegido, en la figura 6.2 se muestra la pantalla de elección de cifrado.

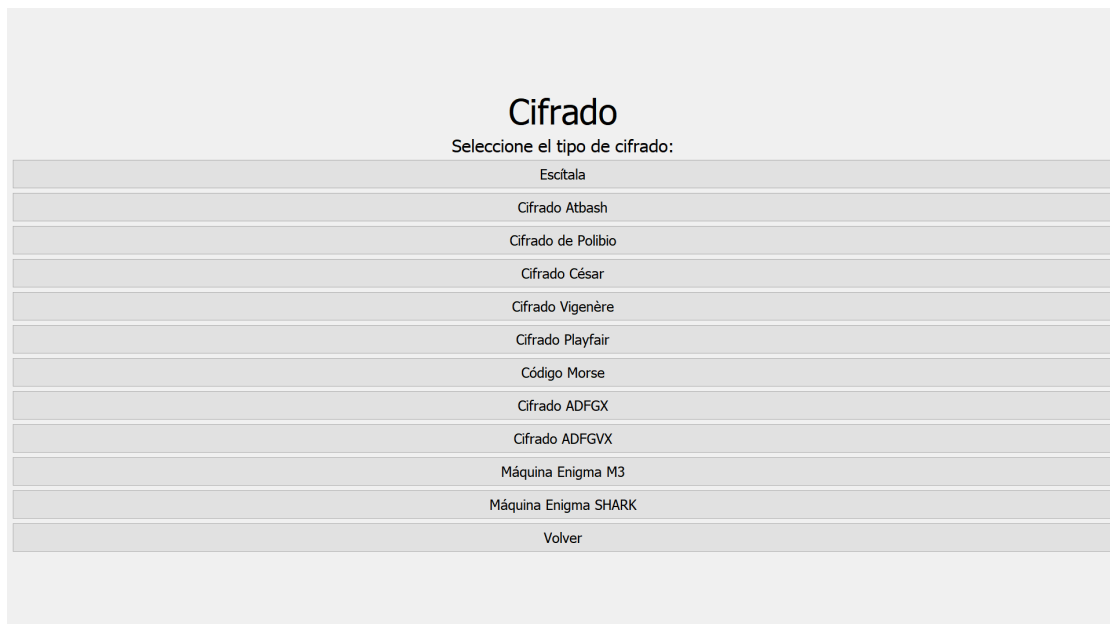


Figura 6.2: Pantalla de Selección de cifrado en la aplicación

Al seleccionar una de las opciones, se le ofrece al usuario información sobre el

sistema con el que va a trabajar y los campos para llevar a cabo la operación. En la figura 6.3 se muestra un ejemplo para el cifrado Vigenère.

Cifrado Vigenère

Información sobre el cifrado Vigenère

Blaise de Vigenère, diplomático francés nacido en 1523, fue el que se llevó la fama final en la revolución de los cifradores polialfabéticos. Este método hace uso de 26 alfabetos y, haciendo uso de una clave de mensaje escrita de forma superpuesta al texto claro, las letras se van cifrando tomando el alfabeto de la letra de la clave y la posición de la letra del texto plano. Esta idea ya había sido propuesta por Bellaso, pero Vigenère introdujo una clave de tabla para modificar la tabla de Tritemio (26 alfabetos en orden alfabético) y crear su propio método. Introduzca el texto claro, una clave de tabla (si quiere cifrar con la tabla de Tritemio no es necesario) y una clave de mensaje para obtener el texto cifrado.

Tabla utilizada:

Texto claro:

Clave de tabla (vacío para cifrar con la Tabla de Tritemio):

Clave de mensaje:

Cifrar

Resultado:

Volver

Figura 6.3: Pantalla de cifrado Vigenère en la aplicación

Finalmente, una vez que el usuario introduce los datos y pulsa sobre el botón de la operación, aparece la tabla utilizada (en el caso en el que el sistema utilice tablas) y el resultado en la casilla de resultado como aparece en la figura 6.4.

Cifrado Vigenère

Información sobre el cifrado Vigenère

Blaise de Vigenère, diplomático francés nacido en 1523, fue el que se llevó la fama final en la revolución de los cifradores polialfabéticos. Este método hace uso de 26 alfabetos y, haciendo uso de una clave de mensaje escrita de forma superpuesta al texto claro, las letras se van cifrando tomando el alfabeto de la letra de la clave y la posición de la letra del texto plano. Esta idea ya había sido propuesta por Bellaso, pero Vigenère introdujo una clave de tabla para modificar la tabla de Tritemio (26 alfabetos en orden alfabético) y crear su propio método. Introduzca el texto claro, una clave de tabla (si quiere cifrar con la tabla de Tritemio no es necesario) y una clave de mensaje para obtener el texto cifrado.

Tabla utilizada:

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

Texto claro:

La felicidad esta en el saber

Clave de tabla (vacío para cifrar con la Tabla de Tritemio):

Clave de mensaje:

Clave

Cifrar

Resultado:

NLFZPKNIYEFPISOEGYEGWCMEM

Volver

Figura 6.4: Pantalla de cifrado Vigenère tras operación

Todas las operaciones de cifrado de los 11 métodos implementados siguen una

Capítulo 6. Resultados y conclusiones

estructura similar. La pantalla de selección de descifrado es idéntica a la pantalla de selección de cifrado (figura 6.5) y el interior sigue el mismo esquema (figura 6.6).



Figura 6.5: Pantalla de Selección de descifrado en la aplicación



Figura 6.6: Pantalla de descifrado Playfair en la aplicación

Por último, los métodos de criptoanálisis siguen el mismo diseño de los métodos de cifrado y descifrado. Un ejemplo con el análisis de frecuencias puede apreciarse en la figura 6.7.

Análisis de frecuencias

Información sobre el análisis de frecuencias

La primera fuente en la que se describe el análisis de frecuencias data del siglo IX y su autor fue Al-Kindi, director de la Casa de la Sabiduría. La técnica consiste en examinar todas las palabras que hay en el diccionario de un determinado idioma y observar cuáles son las letras que se repiten con mayor frecuencia. Luego, se lleva a cabo el mismo estudio con las letras o símbolos del texto cifrado, sustituyendo unas por otras basándose en las equivalencias en la frecuencia de aparición. A medida que aumenta la longitud del texto encriptado, la precisión y fiabilidad del cálculo de frecuencias aumentan, lo que hace que el proceso sea más efectivo. Cinco siglos más tarde llegó el análisis de frecuencia al occidente de Europa como método de ruptura de las cifras y los códigos. Esta técnica sencilla y accesible de origen árabe sentó las bases que siguieron un gran número de criptoanalistas para cambiar el transcurso de la historia. Para llevar a cabo un análisis introduzca el texto cifrado y aparecerá el resultado según las frecuencias en el idioma español.

Texto cifrado:

Atacar

Resultado:

Volver

Figura 6.7: Pantalla de análisis de frecuencias en la aplicación

En total, se han codificado once métodos de cifrado (a los que se suman los once correspondientes de descifrado) y dos métodos de criptoanálisis. Esto resulta en un total de veinticuatro pantallas distintas con las que el usuario puede interactuar y conocer más sobre la criptografía.

Como posibles mejoras, se sugiere optimizar la interfaz gráfica con el fin de aumentar la intuitividad y mejorar el diseño estético, lo que mejoraría significativamente la experiencia del usuario. Estas mejoras podrían incluir una disposición más clara y organizada de los elementos, así como el uso de colores e iconos que faciliten la navegación y comprensión de las funcionalidades. Por otro lado, sería beneficioso incorporar más sistemas, especialmente en la sección de criptoanálisis. Esto permitiría a los usuarios explorar una gama más amplia de técnicas y métodos, proporcionando un conocimiento más profundo y completo de la criptografía.

Capítulo 7

Análisis de impacto

El impacto de este Trabajo de Fin de Grado es notable en los siguientes aspectos:

A nivel personal, el impacto se manifiesta en la obtención de un conocimiento más profundo sobre la criptografía que ha despertado en mí un gran interés por continuar investigando y formándome en esta disciplina. La revisión e investigación realizadas me han permitido conocer la importancia estratégica de la criptografía en conflictos bélicos, un aspecto que a menudo no recibe la atención que merece pero que ha desempeñado un papel crucial en la historia. Además, esta investigación me ha brindado la oportunidad de aplicar los conceptos teóricos estudiados en la asignatura de Estructuras Algebraicas en un contexto práctico y real, incrementando mi apreciación por la asignatura y permitiéndome, dos años después, valorar plenamente su importancia.

A nivel social, el trabajo permite ofrecer al lector una visión general de la evolución de la criptografía, permitiéndole comparar la complejidad de los métodos a medida que se producía un avance en el tiempo. También se dan a conocer muchos hitos matemáticos e históricos relacionados con la criptografía a los que habitualmente no se les da la relevancia y la importancia que les corresponde, intentando aportarles una mayor visibilidad para los estudiantes de la escuela. La creación de una aplicación con pequeñas explicaciones y que permite llevar a cabo operaciones criptográficas puede generar un impacto en la educación, ofreciendo una alternativa a los libros, textos históricos y artículos que abordan los mismos sistemas.

Este trabajo también tiene una relación directa con el cuarto y el noveno punto de los Objetivos de Desarrollo Sostenible (ODS) [29] que busca mejorar la calidad educativa y la industria, la innovación y la infraestructura respectivamente.

El cuarto objetivo de los ODS trata de asegurar una educación inclusiva y equitativa de calidad, promoviendo oportunidades de aprendizaje para todos. Al proporcionar una herramienta educativa interactiva que facilita la comprensión de la criptografía, se fomenta el interés y la accesibilidad al conocimiento en este campo, ayudando a estudiantes y educadores a explorar conceptos complejos de manera más práctica y atractiva.

Capítulo 7. Análisis de impacto

El noveno objetivo se enfoca en construir infraestructuras resilientes, promover la industrialización sostenible y fomentar la innovación. Al incorporar tecnologías y métodos innovadores en la enseñanza de la criptografía, se contribuye al avance de la educación STEM (Science, Technology, Engineering y Mathematics), promoviendo una cultura de innovación que es esencial para el desarrollo industrial y tecnológico.

Bibliografía

- [1] M. J. Prieto, *Historia de la criptografía*. La esfera de los libros, 2020.
- [2] L. H. Encinas, *La criptografía*. Los Libros de la Catarata, 2016.
- [3] N. Geographic, *Matemáticos, espías y piratas informáticos*. RBA, 2010.
- [4] J. M. S. Muñoz, “Criptología nazi. los códigos secretos de hitler,” *Revista de investigación Pensamiento Matemático*, 2013.
- [5] D. A. Brihuega, *Criptografía sin secretos con Python*. RA-MA Editorial, 2017.
- [6] F. de Rojas, *La Celestina*. Editorial Porrúa, 1499.
- [7] N. Geographic. (2023) Jerjes, el gran rey de los persas. [Online]. Available: https://historia.nationalgeographic.com.es/a/jerjes-gran-rey-persas_19887
- [8] U. de Granada. (2010) La escítala. [Online]. Available: <https://www.ugr.es/~anillos/textos/pdf/2010/EXPO-1.Criptografia/02a22.htm>
- [9] A. N. de Catalunya. (2016) Nomenclátor criptográfico de Felipe II. [Online]. Available: <https://anc.gencat.cat/es/detall/article/Document-del-mes-Desembre>
- [10] M. tu realidad. (2015) La rejilla del espía. [Online]. Available: <https://matematizaturealidad.blogspot.com/2015/09/la-rejilla-del-espia.html>
- [11] N. Geographic. (2023) Masones y masonería: leyendas, conspiraciones y realidad. [Online]. Available: <https://www.nationalgeographic.es/historia/masones-masoneria-quienes-son-verdad-leyendas-conspiraciones>
- [12] Numerentur. (2018) Aproximación histórica. [Online]. Available: <https://numerentur.org/seguridad-aproximacion-historica/>
- [13] M. Macho. (2013) El telégrafo de Chappe. [Online]. Available: <https://ztfnews.wordpress.com/2013/12/25/el-telegrafo-optico-de-claude-chappe/>
- [14] C. H. M. Ralph Simpson. (2017) Before Enigma: Breaking the Hebern rotor machine. [Online]. Available: <https://computerhistory.org/blog/before-enigma-breaking-the-hebern-rotor-machine/>

- [15] N. Geographic. (2023) Alan turing, el arma secreta de los aliados. [Online]. Available: https://historia.nationalgeographic.com.es/a/alan-turing-arma-secreta-aliados_16352
- [16] Wikipedia. (2024) Arthur scherbius. [Online]. Available: https://es.wikipedia.org/wiki/Arthur_Scherbius
- [17] L. cripta del hacker. (2021) Criptografía en python: Máquina enigma. [Online]. Available: <https://lacriptadelhacker.wordpress.com/2021/01/24/criptografia-en-python-maquina-enigma/>
- [18] F. R. Villatoro. (2012) Nota dominical: Los matemáticos polacos, alan turing y el secreto de la máquina enigma. [Online]. Available: <https://francis.naukas.com/2012/10/28/nota-dominical-los-matematicos-polacos-alan-turing-y-el-secreto-de-la-maquina-enigma/>
- [19] J. H. Silverman, *Abstract Algebra: An Integrated Approach*. American Mathematical Society, 2022.
- [20] E. S. Richard Klima, Neil Sigman, *Applied Abstract Algebra with Maple and Matlab*. CRC Press, 2022.
- [21] M. G. Larragan. (2015) Criptografía xi: La máquina enigma. [Online]. Available: <https://mikelgarcialarragan.blogspot.com/2015/06/criptografia-xi-la-maquina-enigma-x.html>
- [22] Álvaro Ibañez. (2022) Una máquina enigma en un catálogo de christie's, probablemente la mejor forma de verla con lujoso detalle. [Online]. Available: <https://www.microsiervos.com/archivo/seguridad/maquina-enigma-christies.html>
- [23] A. Turing, *¿Puede pensar una máquina?* KRK Ediciones, 2012.
- [24] Wikipedia. (2024) Gordon welchman. [Online]. Available: https://en.wikipedia.org/wiki/Gordon_Welchman
- [25] —. (2024) Código lorenz. [Online]. Available: https://es.wikipedia.org/wiki/C%C3%B3digo_Lorenz
- [26] D. L. Hunsberger, *Biography of Professor Tutte*. University of Waterloo, 2002.
- [27] T. N. M. of Computing. (2019) Colossus. [Online]. Available: <https://www.tnmoc.org/colossus>
- [28] PythonGUIs. (2024) The complete pyqt5 tutorial — create gui applications with python. [Online]. Available: <https://www.pythonguis.com/pyqt5-tutorial/>
- [29] O. de las Naciones Unidas. (2015) Objetivos de desarrollo sostenible. [Online]. Available: <https://www.un.org/sustainabledevelopment/es/objetivos-de-desarrollo-sostenible/>

Anexos



Digital Receipt

This receipt acknowledges that **Turnitin** received your paper. Below you will find the receipt information regarding your submission.

The first page of your submissions is displayed below.

Submission author: DAVID GONZALEZ LABRADA
Assignment title: Turnitin Memoria Final (Moodle PP)
Submission title: TFG Memoria Final David González.pdf
File name: 17896_DAVID_GONZALEZ_LABRADA_TFG_Memoria_Final_Davi...
File size: 15.79M
Page count: 130
Word count: 40,179
Character count: 205,524
Submission date: 03-Jun-2024 12:10PM (UTC+0200)
Submission ID: 2394498167



Este documento esta firmado por



Firmante	CN=tfgm.fi.upm.es, OU=CCFI, O=ETS Ingenieros Informaticos - UPM, C=ES
Fecha/Hora	Mon Jun 03 17:18:01 CEST 2024
Emisor del Certificado	EMAILADDRESS=camanager@etsiinf.upm.es, CN=CA ETS Ingenieros Informaticos, O=ETS Ingenieros Informaticos - UPM, C=ES
Numero de Serie	561
Metodo	urn:adobe.com:Adobe.PPKLite:adbe.pkcs7.sha1 (Adobe Signature)