



Centro Universitario de la Defensa en la Escuela Naval Militar

TRABAJO FIN DE GRADO

Criptografía: De la cifra del César al RSA

Grado en Ingeniería Mecánica

ALUMNO: Iván Baena Gallardo

DIRECTORES: Rodrigo Mariño Villar

CURSO ACADÉMICO: 2023-2024

Universida_{de}Vigo



Centro Universitario de la Defensa en la Escuela Naval Militar

TRABAJO FIN DE GRADO

Criptografía: De la cifra del César al RSA

Grado en Ingeniería Mecánica
Intensificación en Tecnología Naval
Infantería de Marina

Universida_deVigo

RESUMEN

A lo largo de la historia de la civilización los diferentes gobiernos han tenido la necesidad de cifrar sus mensajes para ocultar cierta información a pueblos rivales. La criptografía, ha sido la herramienta empleada para tal fin. A su vez, el conocimiento de esta información ha proporcionado en no pocas ocasiones una posición ventajosa respecto del enemigo, permitiendo la toma correcta de decisiones políticas y la elección adecuada de líneas de acción de carácter militar. Es aquí donde cobra gran importancia la ciencia del criptoanálisis.

El objetivo de este Trabajo de Fin de Grado consiste en el estudio de los diferentes sistemas criptográficos, así como de la evolución de los mismos a lo largo del tiempo. Para ello se analizarán, en primer lugar los sistemas pertenecientes a la criptografía clásica, y posteriormente los criptosistemas modernos.

Este trabajo toma un papel especialmente relevante en una era en la que la sociedad se encuentra intercomunicada en un grado inimaginable años atrás y la criptografía no es una ciencia circunscrita a los estamentos más elevados de poder, sino que constituye una realidad patente en el quehacer cotidiano del ciudadano de a pie.

PALABRAS CLAVE

Criptografía, Información, Criptoanálisis, Criptosistema, Programación.

AGRADECIMIENTOS

Me gustaría agradecer a mi tutor, D. Rodrigo Mariño Villar, su dedicación a lo largo de estos meses. Muchas gracias por ayudarme a desarrollar este trabajo.

CONTENIDO

Contenido.....	1
Índice de Figuras.....	3
Índice de Tablas.....	4
Índice de Códigos	5
1 Introducción y objetivos.....	7
1.1 Preámbulo.....	7
1.2 Objetivos	7
1.3 Organización de la memoria	8
2 Estado del arte.....	9
2.1 Software Sagemath.....	9
2.1.1 Introducción.....	9
2.1.2 Sage como herramienta matemática	9
2.1.3 Historia y motivación	9
2.2 Evolución histórica de la criptografía	10
2.2.1 Primeros mensajes secretos.....	10
2.2.2 Criptografía clásica. De la Antigua Grecia al Renacimiento	10
2.2.3 Criptografía clásica. La Cifra Indescifrable	12
2.2.4 Criptografía clásica durante el siglo XIX.....	13
2.2.5 El salto a la criptografía moderna.....	14
2.2.6 La aparición del cifrado asimétrico	16
2.3 Clasificación de los criptosistemas desarrollados en el presente trabajo	16
3 Preliminares.....	19
3.1 Asociación alfabética.....	19
3.2 Redundancias del lenguaje.....	19
3.3 Códigos de carácter general para el cifrado y descifrado	20
3.4 Números primos y Divisibilidad.....	22
3.5 Máximo común divisor.....	22
3.6 Aritmética Modular	23
3.7 Algoritmo de Euclides	23
3.8 Algoritmo de Euclides extendido	24
3.9 Función ϕ de Euler.....	25
3.10 Teorema de Euler	25
3.11 Teorema Chino de los Restos.....	25

3.12 Pequeño Teorema de Fermat.....	26
4 Criptografía de clave simétrica	27
4.1 Cifra del César	27
4.2 Cifra de Vigenère	29
4.3 Criptosistema de Hill.....	32
4.4 Cifrado por transposición o permutación	36
5 Criptografía de clave pública.....	37
5.1 Criptosistema RSA.....	37
5.2 Firma digital.....	40
5.3 Ataques al algoritmo RSA	40
5.3.1 Ataque por factorización del módulo.....	40
5.3.2 Ataque por módulo común.....	41
5.3.3 Ataque debido a un número e pequeño.....	41
5.3.4 Ataque cíclico	41
6 Criptografía cuántica	43
7 Conclusiones y líneas futuras.....	45
7.1 Conclusiones.....	45
7.2 Líneas futuras.....	45
7.2.1 Criptografía aplicada a defensa	45
7.2.2 Criptografía cuántica	46
8 Bibliografía.....	47
Anexo I: Implicaciones Sociales, Económicas y Ambientales	49
Anexo II: Reflexiones Éticas y Sociales.....	50
Anexo III: Códigos	51

ÍNDICE DE FIGURAS

Figura 2-1 Clasificación de los criptosistema clásicos (autoría propia)	11
Figura 2-2 Escítalo espartano [8]	11
Figura 2-3 Disco de Alberti [11].....	13
Figura 2-4 Máquina analítica de Babbage [13]	14
Figura 2-5 Máquina Enigma [14].....	15
Figura 2-6 Clasificación de los criptosistemas tratados en el trabajo (autoría propia)	17
Figura 3-1 Asociación entre números y caracteres del alfabeto (autoría propia)	19
Figura 3-2 Frecuencia relativa de los monogramas en castellano (autoría propia)	20
Figura 3-3 Frecuencia de uso de cada monograma (autoría propia)	21
Figura 3-4 Proceso de encriptado y desencriptado de mensajes (autoría propia).....	22
Figura 3-5 Iteraciones para calcular el MCD de 195 y 52 (autoría propia)	24
Figura 4-1 Alfabeto cifrado de Julio César (autoría propia).....	27
Figura 4-2 Ejemplo de cifrado utilizando el sistema de Vigenère (autoría propia).....	29
Figura 4-3 Tabla de Vigenère [9].....	29
Figura 4-4 Descifrado del mensaje restando la clave SUNTZU (autoría propia).....	30
Figura 4-5 Obtención de los textos relativos a los caracteres de la clave (autoría propia).....	31
Figura 4-6 Regla EAOS aplicada al texto que se pretende criptoanalizar (autoría propia).....	32
Figura 4-7 Vectores unitarios (autoría propia)	34
Figura 4-8 Vectores unitarios representados por letras (autoría propia).....	34
Figura 4-9 Comparación de los textos y búsqueda de vectores unitarios (autoría propia).....	35
Figura 5-1 Esquema del funcionamiento de los códigos de clave pública (autoría propia)	37
Figura 5-2 Probabilidad de que se produzca una colisión en la paradoja de cumpleaños [9]	42
Figura 6-1 Esquema del protocolo QKD (autoría propia).....	44
Figura 7-1 JTAC del US Air Force realizando el guiado terminal de una aeronave a través de enlace VHF [27].....	46

ÍNDICE DE TABLAS

Tabla 1 Equivalencia alfabeto ASCII (autoría propia)..... 38

ÍNDICE DE CÓDIGOS

A III: Código 1 Asociación 1.....	51
A III: Código 2 Asociación inversa 1	52
A III: Código 3 Contador de letras.....	53
A III: Código 4 Algoritmo de Euclides.....	53
A III: Código 5 Algoritmo de Euclides Extendido	54
A III: Código 6 Cifra del César.....	54
A III: Código 7 Descifrado César.....	55
A III: Código 8 Criptoanálisis César por fuerza bruta	55
A III: Código 9 Cifra de Vigenère.....	56
A III: Código 10 Descifrado Vigenère	56
A III: Código 11 Método Kasiski.....	57
A III: Código 12 Cifrado de Hill.....	58
A III: Código 13 Cálculo matriz inversa.....	59
A III: Código 14 Descifrado Hill	60
A III: Código 15 Cifrado por permutación.....	61
A III: Código 16 Cálculo permutación inversa	61
A III: Código 17 Asociación 2.....	62
A III: Código 18 Selección de primos p y q.....	63
A III: Código 19 Cifrado RSA.....	63
A III: Código 20 Cálculo de la clave privada d.....	64
A III: Código 21 Descifrado RSA.....	64
A III: Código 22 Asociación 2 inversa	65
A III: Código 23 Firma Digital	65
A III: Código 24 Firma digital comprobación.....	66

1 INTRODUCCIÓN Y OBJETIVOS

1.1 Preámbulo

A lo largo de la historia, el ser humano siempre ha tratado de mantener sus secretos a salvo. Así mismo, se las ha ingeniado para ser capaz de transmitir estos secretos a través de mensajes, utilizando diversos mecanismos y técnicas, de tal forma que en caso de que dicho mensaje no llegara a su destino, o fuera interceptado, solo el emisor y el receptor fueran capaces de conocer su verdadero significado.

El empleo y desarrollo de estos métodos para mantener oculto el contenido de un mensaje es lo que se denomina como la ciencia de la criptología. La palabra criptología viene de las raíces griegas *kryptós* y *logos*. La primera de ellas significa oculto, y la segunda hace referencia al conocimiento, la palabra, el estudio o la razón. Por ello, se deduce que la criptología es la ciencia que abarca el conocimiento de lo oculto, concretamente, de aquellos mecanismos que permiten establecer lenguajes o sistemas de comunicación secretos.

La criptología se divide a su vez en dos ramas, la criptografía y el criptoanálisis. Atendiendo a la definición de la palabra criptografía proporcionada por la Real Academia Española se entiende esta como “Arte de escribir con clave secreta o de un modo enigmático” [1]. Sin embargo, en la actualidad, cuando la criptografía comprende el intercambio de claves, la firma digital, protocolos de autenticación de usuario, etc.; esta descripción queda obsoleta, y es necesario hablar de criptografía moderna. En su libro *Introduction to Modern Cryptography*, los profesores Jonathan Katz y Yehuda Lindell definen la criptografía moderna como “El estudio de las técnicas matemáticas para proporcionar seguridad a la información digital, sistemas y computación distribuida contra ataques adversarios” [2]. El criptoanálisis en contraposición es la rama que estudia como romper códigos y criptosistemas.

La criptografía es por ende, tan antigua como lo es la capacidad de comunicación, y su evolución está ligada a la del ser humano. La competencia de enviar mensajes ocultos, y sobre todo, la habilidad para descifrarlos han influido de forma considerable en el desarrollo de la historia. De hecho, la evolución del conocimiento humano es la que ha determinado la seguridad de los códigos de encriptado, de tal forma que en una época donde solo algunos hombres eran capaces de leer y escribir, era suficiente con realizar alguna pequeña alteración ortográfica para que el mensaje quedara a salvo. Así nacieron los primeros códigos, y su desarrollo a dado lugar a los algoritmos utilizados hoy en día.

1.2 Objetivos

El principal objetivo de este trabajo consiste en trazar la evolución de los códigos y criptosistemas. Para ello, en primer lugar se estudiarán los criptosistemas clásicos, que asientan las bases y proveen cierto trasfondo, lo que simplifica la tarea de comprender los criptosistemas actuales. Posteriormente se procederá al estudio de los criptosistemas modernos, analizando con mayor detenimiento el

criptosistema RSA (Rivest, Shamir, Adleman), considerado el primer sistema de cifrado de clave pública y uno de los más relevantes.

Además, la palabra evolución resulta verdaderamente pertinente, ya que en definitiva, el desarrollo de la criptografía puede definirse como un conflicto permanente entre cifrado y criptoanálisis, que da lugar a la aparición de nuevas formas de encriptar mensajes, cada cual más compleja que la anterior.

El segundo objetivo de este trabajo es manifestar que la criptografía no es una ciencia sujeta al estudio de los matemáticos y criptoanalistas, sino que en esta época, definida como “Era de la información” [3] por el sociólogo Manuel Castells Oliván en su trilogía homónima, el proceso de cifrado y codificación desempeña un papel cada vez más importante, presente en los actos más ordinarios del día a día como llamar por teléfono, enviar un correo electrónico, comprar de manera telemática, visitar las redes sociales o consultar el fondo bancario.

Además, aunque actualmente la criptografía es una realidad patente en la vida civil, esto no quiere decir que constituya un concepto latente en el ámbito militar. Al contrario, el intercambio secreto de información ha sido siempre una necesidad significativa en el entorno castrense, y prueba de ello es la existencia de obras como el *Tratado de criptografía con aplicaciones especial al Ejército* [4] perteneciente al teniente L.G. Carmona o *La cryptographie militaire* [5] de Auguste Kerchoffs, y lo es hoy más que nunca, en un momento histórico en el que el arma principal de la guerra es la información. De hecho, por este motivo, en 2021 el Ministerio de Defensa creó la Jefatura de Sistemas Satelitales y de Ciberdefensa (JSSAT-CIBER). [6]

1.3 Organización de la memoria

En relación con la organización de la memoria esta consta de seis apartados y tres anexos. Los apartados por orden son los siguientes: Estado del arte, Preliminares, Criptografía de clave simétrica, Criptografía de clave pública, Criptografía cuántica y Conclusiones y líneas futuras.

En el Estado del arte, se realiza un breve recorrido histórico sobre la evolución de los códigos, con mayor detenimiento en aquellos más relevantes y que constituirán los apartados posteriores. Además, se referencian aquellas publicaciones y trabajos que influyeron de forma significativa en el desarrollo de la criptografía y se introducen términos referentes al ámbito de la criptografía.

En el apartado de Preliminares se presentan ciertas consideraciones relativas al lenguaje que permitirán el criptoanálisis de los códigos empleados, así como los principios matemáticos detrás del funcionamiento de cada uno de ellos.

En el siguiente apartado, Criptografía de clave simétrica, se realiza el estudio de la Cifra del César, la Cifra de Vigenère, Criptosistema de Hill y el Cifrado por transposición o permutación. Para cada uno de ellos corresponde su explicación matemática, el cifrado de un mensaje junto con la programación de dicho proceso, así como del descifrado conociendo las claves pertinentes, y por último el criptoanálisis del sistema, es decir, el ataque del sistema de cifrado.

De la misma manera, en el apartado Criptografía de clave pública, se realiza el mismo proceso para el Criptosistema RSA. Además se analiza el proceso de Firma digital, así como los Ataques al algoritmo RSA.

A continuación, en el apartado Criptografía cuántica, se indaga acerca de la computación cuántica y el proceso de distribución de claves, así como de la repercusión de los mismos.

Por último, en Conclusiones y líneas futuras se exponen cuales podrían ser las líneas de investigación futuras para un trabajo subsiguiente.

Además se incluyen tres anexos: el Anexo I: Implicaciones Sociales, Económicas y Ambientales, el Anexo II: Reflexiones Éticas y Sociales y el Anexo III: Códigos, donde se adjuntan los códigos mencionados y citados a lo largo de la memoria.

2 ESTADO DEL ARTE

2.1 Software Sagemath

2.1.1 Introducción

En el apartado de Desarrollo de este trabajo, cada uno de los criptosistemas que se analizan es programado en lenguaje Python e implementado en el software Sagemath. Por ello, el primer subapartado del Estado del Arte va destinado a introducir esta herramienta.

2.1.2 Sage como herramienta matemática

Sage es un software de cálculo simbólico capaz de implementar algoritmos matemáticos en diferentes contextos. Puede utilizarse como calculadora científica, y es capaz de manipular diferentes tipos de números como enteros, racionales, aproximaciones numéricas, complejos de precisión arbitraria o elementos de campos finitos. Además de ejecutar operaciones numéricas, Sage puede utilizarse como un sistema de álgebra computacional, y por tanto permite resolver ecuaciones lineales, factorizar, simplificar y realizar operaciones en anillos arbitrarios de polinomios o campos de funciones racionales entre otras. También maneja expresiones en análisis, álgebra lineal, probabilidad, estadística y combinatoria. En resumen, proporciona un acceso sencillo y consistente a una amplia gama de características matemáticas, sin necesidad de transferir datos entre distintos softwares o aprender diversos lenguajes de programación.

El lenguaje de programación utilizado por Sage para emitir comandos es Python. Además, incluye la capacidad de soportar notaciones matemáticas comunes en uso interactivo. Los usuarios tienen la posibilidad de escribir fragmentos de código o programas cuando necesitan realizar cálculos complejos o repetitivos en lugar de comandos simples de una sola línea. En caso de que estos sean de interés general, pueden ser enviados para su inclusión en Sage, creando funciones y bibliotecas públicas.

2.1.3 Historia y motivación

Los inicios de este software se remontan a 2005 cuando el académico estadounidense William Stein, comenzó el proyecto Sage con la intención de crear un software gratuito para realizar cálculos matemáticos, desarrollado por usuarios y para usuarios. Esta iniciativa fue un éxito, y pronto arraigó una comunidad internacional con cientos de desarrolladores alrededor de Sage, la mayoría de ellos, profesores e investigadores. En una primera instancia, Sage estaba orientado hacia la teoría de números, sin embargo, debido a las contribuciones, sus capacidades se extendieron y abarcó nuevas áreas matemáticas.

Sage puede ser utilizado y descargado de forma gratuita. Además es un software libre, y por tanto, los autores no imponen restricciones en su uso, distribución, estudio o modificación. Para utilizar Sage,

solo se necesita un buscador web. El software puede utilizarse tanto en línea, por ejemplo a través del portal COCALC, sin necesidad de instalar ningún programa en el dispositivo, como sin conexión, descargando el programa SageMath. Este programa está disponible para la mayoría de sistemas operativos: Linux, Windows y MacOS. [7]

2.2 Evolución histórica de la criptografía

2.2.1 Primeros mensajes secretos

El primer vestigio del uso de mensajes secretos se remonta a la Antigua Grecia. En la obra *Historias*, cuyo origen se estima alrededor del año 430 a.C., el historiador Heródoto de Halicarnaso narra el conflicto bélico entre griegos y persas. Él mismo, afirma que fue el arte de la criptografía lo que salvó a Grecia de ser conquistada por el líder persa Jerjes.

El conflicto comienza en el año 480 a.C., cuando el Imperio aqueménida orquesta la invasión de Atenas y Esparta como represalia ante su falta de pleitesía. El ataque debía tomar por sorpresa los griegos, sin embargo, la movilización de tropas no había pasado desapercibida a Deramato, un griego exiliado en la ciudad persa de Susa, quien decidió advertir a los espartanos de la amenaza inminente que se cernía sobre ellos. La complejidad de la empresa radicaba en como ingeniárselas para lograr enviar el mensaje sin que este fuera interceptado.

Según Herodoto, el método consistió en lo siguiente: se retiró la cera de una tabla de madera, donde se serigrafió el mensaje, y posteriormente se cubrió de nuevo. A pesar de que el mensaje fue interceptado por guardas persas durante el camino, pasó inadvertido debido a que aparentemente la tabla se encontraba en blanco. Finalmente, la tabla llegó a su destino y los griegos utilizaron esta ventaja y urdieron una estratagema para tender una emboscada a los persas en la bahía de Salamina.

En la historia anterior, se ocultó el mensaje y se logró que solo el emisor y el receptor conocieran su significado, sin embargo, este método de cifrado no puede considerarse como criptográfica, ya que esta consistiría en ocultar el significado pero no el mensaje. En el caso anterior, la técnica empleada se conoce como esteganografía, y consiste en hacer pasar desapercibido el mensaje a los ojos del público no deseado. La técnica más conocida dentro de la esteganografía es la escritura con tinta invisible.

Aunque la criptografía y la esteganografía son independientes, pueden utilizarse de forma conjunta para ocultar un mensaje. Por ejemplo, en la Segunda Guerra Mundial, los agentes secretos alemanes en América del Sur, reducían una página de texto correspondiente a un mensaje cifrado a un cuadrado de dimensiones milimétricas que ocultaban en un mensaje en claro, pretendiendo hacer creer que se trataba de un punto y aparte cualquiera. No fue hasta 1941 cuando el FBI descubrió uno de estos misteriosos puntos. [8]

2.2.2 Criptografía clásica. De la Antigua Grecia al Renacimiento

La criptografía puede dividirse en dos etapas históricas: la criptografía clásica y la criptografía moderna. La criptografía clásica abarca aquellos códigos de encriptado empleados hasta la mitad del siglo XX. Estos a su vez pueden clasificarse según el sistema utilizado en cifradores por transposición y cifradores por sustitución. Los primeros cifran el mensaje cambiando la posición de los caracteres dentro del texto, y los segundos utilizan un alfabeto paralelo inventado para tal fin, de tal forma que se sustituye cada carácter original por su homólogo. En la siguiente imagen aparecen clasificados los sistemas de cifrado más relevantes en la historia de la criptografía clásica.

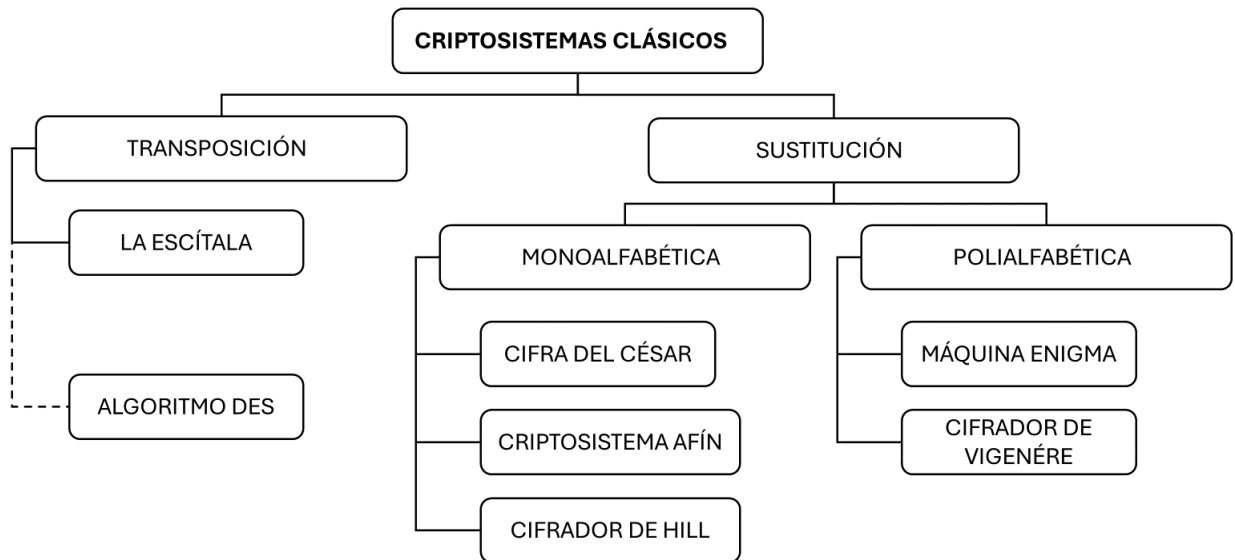


Figura 2-1 Clasificación de los criptosistema clásicos (autoría propia)

El primer aparato criptográfico militar de la Historia es la escítala, también conocida como el escítalo espartano, el cual data del siglo V a. C. Este sistema consistía en una cinta de cuero, la cual se enrollaba alrededor de un bastón y se escribía el mensaje en claro a lo largo de ella. Una vez escrito, se desenrollaba y era entregada a un mensajero. Para leer el mensaje se necesitaba otro bastón con el mismo diámetro que aquel que había sido empleado para redactar el mensaje. En caso de ser interceptado, el mensaje carecía de sentido. Como curiosidad, de este hecho proviene la expresión “ostentar el bastón de mando”. Este sistema, es el primer ejemplo del cual se tiene constancia que utilice el cifrado por permutación o transposición.



Figura 2-2 Escítalo espartano [8]

Por otro lado, el primer empleo de un sistema de cifrado por sustitución se atribuye al dictador Julio César, quien utilizaba el cifrado de información para transmitir mensajes en sus campañas. En este sistema cada letra del alfabeto latino es sustituida por otra, de tal forma que cada letra se corresponde de forma unívoca con otra que la sustituye.

El tipo de cifra utilizada por Julio César es lo que se conoce actualmente como cifrado por sustitución monoalfabética. Esto se debe a que a cada carácter del alfabeto en claro, corresponde un solo carácter del alfabeto utilizado para ocultar el mensaje. Es aquí precisamente donde se encuentra el eslabón débil de esta cifra. Esto se debe a que al sustituir cada letra por un solo símbolo o letra, aunque el texto cifrado resulte incompresible, este mantiene intactas la propiedades referentes a la redundancia del lenguaje. Es decir, dicho símbolo se repetirá el mismo número de veces y en la misma posición que en el texto en claro. Se indagará más acerca de la redundancia del lenguaje al inicio del desarrollo, sin embargo, debe mencionarse en este apartado de la memoria ya que con el estudio de dichas propiedades comenzó el criptoanálisis. [9]

Se estima que los primeros criptoanalistas surgieron en siglo VIII, en el seno de la civilización islámica, en una época en la cual la expansión del conocimiento, y concretamente de la divulgación de la escritura se encontraba en auge. El estudio de la frecuencia con que aparecían diferentes palabras en el texto vino motivado por el estudio del Corán. Los teólogos trataban de demostrar mediante la consistencia de las estructuras gramaticales y la etimología de las palabras, que en efecto las escrituras que analizaban eran atribuibles al profeta Mahoma. Para ello, los analistas buscaban en el texto palabras que habían evolucionado recientemente, para situarlo cronológicamente en un momento posterior a la revelación. Aunque no se tiene constancia de quien fue el primer autor en atacar satisfactoriamente un criptograma, las técnicas criptoanalíticas aparecen descritas por primera vez en una obra del siglo IX escrita por el filósofo Al Kindi. [10]

El desarrollo intelectual que experimentó Oriente Medio se hizo esperar en Europa hasta el Renacimiento. Sin embargo, el estudio de la criptografía se desarrolló también en Occidente, concretamente en el interior de los monasterios. Esta labor fue llevada a cabo por los monjes copistas medievales, quienes intrigados por los términos cifrados que encontraban en el Antiguo Testamento, progresaron en el estudio de esta materia. Este criptosistema al que se enfrentaban los monjes era el *Atbash*, un mecanismo tradicional de cifrado por sustitución hebreo. Aunque se desconoce, aparentemente la intención de introducir palabras cifradas era dotar de un aura de intriga a la Escritura, más que ocultar su significado. Independientemente del fin con que se realizara, la existencia de criptogramas provocó que la criptografía proliferara en Europa. El primer libro que describe el uso de códigos secretos en Europa es la *Epístola sobre las artes secretas y la nulidad de la magia*, escrita por el filósofo inglés y monje franciscano Roger Bacon.

Durante los siglos XIV y XV, la criptografía continuó extendiéndose en Europa, anidando en los escritos de alquimistas, inventores y científicos que hacían uso de ella para mantener sus descubrimientos a salvo. Además, con la llegada del Renacimiento y el florecimiento de la diplomacia, el lenguaje secreto comenzó a utilizarse también en el ámbito y político. Sin embargo, el desarrollo de la criptografía incentivó inexorablemente el desarrollo del criptoanálisis y el cifrado por sustitución monoalfabética dejó de ser seguro. Era el momento de que surgiera un nuevo método de cifrado. [11]

2.2.3 Criptografía clásica. La Cifra Indescifrable

El camino hacia el cifrado polialfabético no transcurrió de la noche a la mañana, fue un proceso de siglos. La primera contribución a este nuevo sistema de cifrado se atribuye al polifacético humanista Leon Battista Alberti. Nacido en Génova en 1404, desarrolló sus conocimientos en las ramas de arquitectura, música, poesía, etc., así como en la criptografía. Trabajó en el Vaticano durante tres papados, y fue precisamente allí donde nacieron los primeros brotes del cifrado por uso de varios alfabetos secretos. Según la tradición, en el transcurso de un paseo por los jardines de la Santa Sede con su amigo Leonardo Dato, secretario papal, este le dejó ver la necesidad del Santo Padre de utilizar un sistema de cifrado para sus mensajes. Sin embargo, todavía no habían conseguido dar con uno que proporcionara las adecuadas garantías.

Alberti se hizo cargo de la petición y desarrolló un código innovador y muy robusto para la época, el cual utilizaba varios alfabetos de cifrado posibles para una misma letra. El autor plasmó su ingenioso método en el ensayo *De Componendis Cyphris*, sin embargo este se mantuvo oculto durante un siglo, siendo solo utilizado por las altas esferas del Vaticano. El sistema de Alberti utilizaba un disco compuesto por anillos concéntricos, uno interior fijo, y uno externo móvil, que al rotar permitía utilizar distintas relaciones entre los caracteres, dando lugar a que un mismo mensaje pudiera estar cifrado de numerosas formas distintas dando lugar a criptogramas sin ningún rasgo común en apariencia. En la figura puede observarse el llamado disco de Alberti, con el que se llevaba a cabo el cifrado.



Figura 2-3 Disco de Alberti [11]

El cifrado polialfabético continuó extendiéndose en el ámbito de la diplomacia y la correspondencia secreta a lo largo del siglo XV. Finalmente, fue el matemático francés Blaise de Vigenère quien popularizó este método, publicándolo en su obra *Traicté des chiffres* en 1586. Lo novedoso del método de Vigenère consistía en establecer un patrón de cifrado de tal forma que con cada nueva letra se utilizaba un alfabeto distinto del anterior. Esto se lograba acordando una palabra clave, que recorría cíclicamente el texto a cifrar. Aunque fue Vigenère quien se llevó la fama, existe una obra anterior titulada *La cifra del Sig. Giovan Battista Belaso*, escrita en 1553 por el autor cuyo nombre aparece en el propio título de la misma, que ya proponía este sistema. [8] [11]

2.2.4 Criptografía clásica durante el siglo XIX

El siglo XIX fue testigo de grandes avances tecnológicos, abarcando el surgimiento y expansión de la Revolución Industrial. Fue en esta época cuando se asentaron las bases de la computación y aparecieron nuevos inventos como el telégrafo o la radio que supondrían un punto de inflexión en los métodos de comunicación. En este contexto, la criptografía no podía, como es lógico, permanecer ajena a estos cambios. Además, en este período los códigos secretos habían ganado bastante popularidad y protagonizaron numerosas obras literarias como *El escarabajo de Oro* de Edgar Allan Poe o el relato de *Los Bailables*, que aparece en *Las aventuras de Sherlock Holmes* escrito por Arthur Conan Doyle.

Merece especial mención en este apartado el trabajo realizado por el científico británico Charles Babbage, quien diseñó una máquina capaz de calcular funciones polinómicas. Posteriormente, escribió varios documentos desarrollando el concepto de la Máquina Analítica, también invención suya. Aunque esta no llegó a construirse, la descripción de la misma se publicó en 1843 bajo el título de *Sketch of the Analytical Engine Invented by Charles Babbage*. La máquina de Babbage es considerada el primer diseño de una computadora programable. Este hecho es relevante por la relación que se establecerá posteriormente entre la criptografía y la computación. En este punto es destacable el trabajo de un ingeniero español, Leonardo Torres Quevedo, quien en 1893 presentó la *Memoria sobre las máquinas algebraicas*, diseñando la primera de tales características en España. [12]

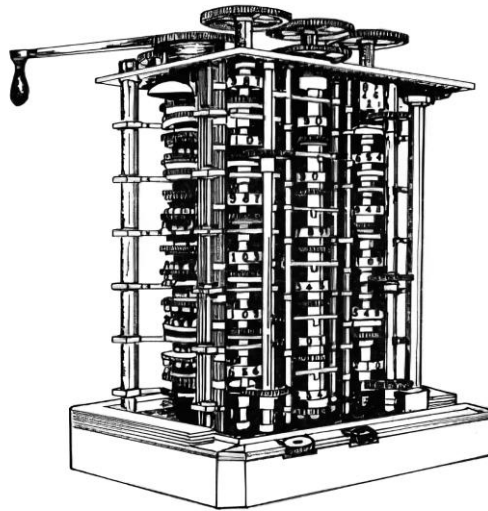


Figura 2-4 Máquina analítica de Babbage [13]

Además de sus avances en el ámbito de la computación, Babbage contribuyó de forma decisiva al desarrollo de la criptografía, ya que fue el quien resolvió la *chiffre indéchiffrable* de Vigenère. Este hecho es el mayor avance en la técnica del criptoanálisis desde que los estudiosos árabes descifrarán el sistema de sustitución monoalfabética. Sin embargo, el trabajo de Babbage no se debe a su conocimiento de las máquinas analíticas, sino a su intuición e interés por la criptografía, unidos a algo de suerte. El criptoanálisis de la Cifra de Vigenère se basa también en las características redundantes del lenguaje, sin embargo resulta algo complejo y se explicará en el desarrollo del trabajo. A pesar de la importancia de los avances de Babbage, estos se mantuvieron en secreto y no fue hasta 1863 cuando el militar prusiano Friederich Willhelm Kasiski hizo público un método muy similar, que había desarrollado de forma paralela.

Por todo lo anterior, el siglo XIX finalizó de forma confusa, en un momento de la historia en el cual no existían métodos de cifrado fiables, y sin embargo, la invención de Marconi demandaba más que nunca la posibilidad de codificar la información. [8]

2.2.5 El salto a la criptografía moderna

Al inicio del siglo XX y durante la Primera Guerra Mundial no se produjeron grandes avances en el descubrimiento de códigos. No obstante, sí que se produjeron hechos relevantes, como la creación de la llamada “Sala 40”, una unidad de inteligencia británica que entre otras hazañas interceptó y descifró el telegrama Zimmermann, revelando el intento de los alemanes de arrastrar a México a la Gran Guerra. Este hecho hizo girar las tornas de la opinión pública estadounidense, y por tanto influyó decisivamente en el resultado del conflicto. El siguiente gran hito ocurriría en 1918, cuando la compañía alemana Scherbius y Ritter, dedicada a la creación de proyectos técnicos de índole diversa, desarrolló una pieza de maquinaria criptográfica denominada Enigma que revolucionaría el ámbito de la criptografía.

La máquina Enigma era en definitiva la mecanización del cifrado polialfabético. Para la cifra de Vigenère se necesitaba una clave que se repitiera de forma cíclica a lo largo del mensaje. Como este proceso se hacía a mano, era necesario que la clave no fuera excesivamente larga para no hacer de la operación de cifrado y descifrado un proceso tedioso. Lo innovador de la máquina Enigma era que permitía utilizar una clave de cifrado tan larga como el mensaje, lo cual dificultaba de manera exponencial criptoanalizar el mismo.

Tras la Gran Guerra los británicos continuaron espionando las comunicaciones alemanas, sin embargo en 1926 lograrían interceptar un mensaje que los dejaría consternados, al no ser capaces de determinar

si quiera como había sido cifrado. Alemania poseía el código más robusto hasta la fecha, y se avecinaba otro gran conflicto.



Figura 2-5 Máquina Enigma [14]

El primer paso hacia la resolución de esta máquina lo dieron los criptoanalistas polacos en la década de 1930, al ser capaces de reconstruir una réplica de la misma y descifrar parcialmente ciertos mensajes. No obstante los alemanes también mejoraron la máquina, que había sido patentada hacía ya más de una década. La tensión creciente en Europa hacía todavía más urgente la labor de desentrañar el secreto, y finalmente el estallido de la Segunda Guerra Mundial en 1939 provocó que los aliados centraran muchos de sus esfuerzos en descifrar las comunicaciones enemigas.

Los británicos fueron los que cargaron con la mayor parte de la iniciativa criptoanalítica por parte de los aliados. En este contexto se reactivó la Sala 40 y algunas de las mentes más brillantes del país fueron contratadas para trabajar en el descifrado de mensajes alemanes. Una de estas personas fue Alan Turing, quien ya había publicado artículos en los cuales describía el funcionamiento de máquinas similares a las de Babbage, capaces de realizar cálculos matemáticos de mayor complejidad. Estos aparatos, denominados máquinas de Turing, fueron los precursores de la máquina “Bombe”, construida en 1940, que debido al elevado número de operaciones lógicas que podía realizar disminuyó drásticamente el tiempo requerido para romper la cifra de Enigma. La resolución de la cifra utilizada por Enigma fue decisiva, aportando una ventaja estratégica significativa a los aliados. [15]

Además de la máquina Enigma, durante la Segunda Guerra Mundial se utilizaron otros métodos menos complejos, pero necesarios para cifrar las transmisiones en el campo de batalla. En la Guerra del Pacífico, los Marines reclutaron indios Navajos como operadores radio, quienes cifraban las comunicaciones en su idioma nativo, el cual carecía de alfabeto escrito y solo era conocido por ellos mismos. [16]

Dejando a un lado la Segunda Guerra Mundial, durante el siglo XX se desarrollaron otros sistemas de cifrado. Uno de estos fue la Cifra de Hill. En 1929, Lester S. Hill publicó un artículo en la revista *The American Mathematical Monthly* [17], en el cual planteaba un nuevo sistema de cifrado que estaba basado en la transformación lineal matricial. Por desgracia, su método era demasiado complejo para ser implementado en una máquina y no pudo competir con Enigma. No obstante, será estudiado en el desarrollo del trabajo.

Otro de los hechos más significativos en la historia de la criptología es la publicación de dos artículos pertenecientes al matemático Claude Shannon al final de la década de los 40, estos artículos son *A Mathematical Theory of Communication* [18] y *Communication Theory of Secrecy Systems* [19]. Este episodio asienta las bases para la aparición de la criptografía moderna. En ellos, Shannon resume los mecanismos de criptología anteriores y propone dos técnicas de encriptado, ambas de clave secreta, a las que llama difusión, consistente en la transposición, y la confusión, basada en la sustitución. A partir

de este momento, las matemáticas adquieren un valor fundamental y empiezan a desarrollarse teorías basadas en los principios de Kerchoffs, recogidos en su libro *La cryptographie militaire* [5]. Los dos principios de Kerchoffs más relevantes son los siguientes:

- Si el sistema no es teóricamente irrompible, al menos debe serlo en la práctica.
- La seguridad del sistema no puede depender de que se mantenga en secreto el algoritmo de cifrado. Sólo debe depender de que se mantengan en secreto las claves. [9] [5]

2.2.6 La aparición del cifrado asimétrico

Tras la publicación de los artículos de Shannon, los sistemas de cifrado continuaron desarrollándose, atendiendo a las técnicas de difusión y confusión descritas por él. La primera, consiste en diluir las propiedades estadísticas del lenguaje mediante la permutación o transposición, y la segunda, en desordenar el mensaje, mediante la sustitución, de tal forma que la relación entre criptograma y mensaje en claro sea lo más compleja posible y no pueda atacarse el algoritmo.

El algoritmo de cifrado DES (Data Encryption Standard) fue uno de los más utilizados y supuso un gran avance en el desarrollo de la criptografía moderna. Este sistema fue desarrollado por IBM en 1973, inicialmente denominado Lucifer, utilizando las rondas de Feistel, y la NSA (National Security Agency) lo adoptó como algoritmo de encriptado para proteger material clasificado del gobierno de EEUU. Actualmente el criptosistema DES es considerado inseguro por el tamaño de la clave que utiliza. En su lugar se emplea el 3DES y el AES (Advanced Encryption Standard).

Sin embargo, la seguridad de estos sistemas reside en que la clave permanezca en secreto, y en ocasiones, no es posible compartir dicha clave entre emisor y receptor a través de un medio seguro. Por ello, es necesario un sistema que permita a los usuarios cifrar mensajes, sin poseer la clave para descifrarlos. Esto es lo que se conoce como sistemas de clave pública.

El siguiente gran hito en la historia de la criptografía es la publicación del artículo *New directions in cryptography* [20], en 1976, escrito por Baile Whitfield Diffie y Martin Hellman, quienes desarrollaron el primer algoritmo asimétrico, conocido más tarde como Diffie-Hellman, en honor a sus descubridores. Este algoritmo surgió de la necesidad de intercambiar contraseñas simétricas secretas a través de un medio no seguro como internet.

Los algoritmos de cifrado asimétrico se basan en funciones trampa, cuyo cálculo es relativamente sencillo, pero sin el conocimiento de toda la información son irreversibles a nivel computacional. Estas funciones son por ejemplo la dificultad de factorizar números muy grandes, el problema del logaritmo discreto y el problema de las raíces cuadradas. [21]

El criptosistema RSA fue desarrollado en 1979 por Ron Rivest, Adi Shamir y Leonard Adleman, y es el primer algoritmo de clave pública, estudiado en el desarrollo. La seguridad de este sistema se basa en la complejidad a nivel computacional de factorizar un número compuesto muy grande. Otro de los algoritmos de cifra pública más usados es el propuesto en 1985 por Taher Elgamal, que basa su fortaleza en la dificultad de resolver logaritmos discretos en un primo grande. [22]

2.3 Clasificación de los criptosistemas desarrollados en el presente trabajo

Una vez contextualizado el trabajo, solo queda proceder con el desarrollo. En la siguiente figura se observa la clasificación de los criptosistemas estudiados en este Trabajo de Fin de Grado.

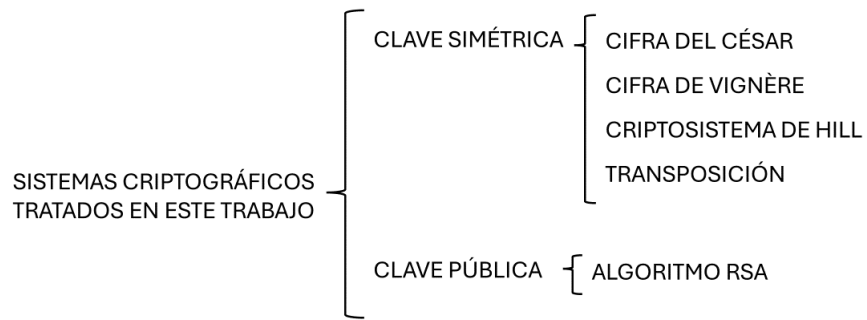


Figura 2-6 Clasificación de los criptosistemas tratados en el trabajo (autoría propia)

3 PRELIMINARES

3.1 Asociación alfabética

Cuando se habla de cifrado clásico, generalmente se utilizan caracteres del mismo alfabeto para cifrar el texto en claro. La forma más sencilla de llevar a cabo esta tarea es relacionar cada letra con un número, de tal forma que para el alfabeto latino moderno existen 26 caracteres, en su forma tanto mayúscula como minúscula, al que se pueden añadir letras adicionales, como es el caso de la ñ en el abecedario castellano.

En función de los caracteres considerados se pueden definir diferentes alfabetos: letras mayúsculas, letras mayúsculas y minúsculas, letras acentuadas, letras y dígitos o el alfabeto ASCII (American Standard Code for Information Interchange) entre otros. En este trabajo se utilizará el alfabeto latino moderno, considerando solo la versión mayúscula de cada letra. Además, no se utilizará el espacio en blanco ya que, a la hora de someter un mensaje cifrado a un criptoanálisis, esto es, tratar de romper el código y descifrar el mensaje, este valor se repite con frecuencia, aproximadamente algo menos del 20%, y proporciona una pista relevante para el criptoanalista. Por tanto, nuestra asociación entre caracteres del alfabeto y números 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
0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5
										1										2					

Figura 3-1 Asociación entre números y caracteres del alfabeto (autoría propia)

3.2 Redundancias del lenguaje

La siguiente consideración que debe tener en cuenta un criptoanalista es la redundancia del lenguaje, es decir, con qué frecuencia se repiten como norma general los caracteres. Esto dependerá del idioma, y del tipo de texto, por ejemplo, en un texto en el que aparezcan muchos verbos en infinitivo habrá muchas “R”, en un texto relacionado con el comercio agrícola seguramente la letra “K” tendrá un peso relevante, debido a la aparición de la palabra “kilo”. En la Biblia -en su traducción castellana- la palabra más repetida es “Dios” que aparece unas 4.700 veces -depende de la traducción-. En castellano, las letras más utilizadas en orden son: E, A, S, O, I, N, R, D y T. En la siguiente tabla aparece la frecuencia relativa de los monogramas en castellano, clasificados en 3 grupos.

Frecuencia Alta		Frecuencia Media		Frecuencia Baja	
E	13,11	C	4,85	Y	0,79
A	10,60	L	4,42	Q	0,74
S	8,47	U	4,34	H	0,60
O	8,23	M	3,11	Z	0,26
I	7,16	P	2,71	J	0,25
N	7,14	G	1,40	X	0,15
R	6,95	B	1,16	W	0,12
D	5,87	F	1,13	K	0,11
T	5,40	V	0,82	Ñ	0,10

Figura 3-2 Frecuencia relativa de los monogramas en castellano (autoría propia)

Además, debe tenerse en cuenta que aproximadamente el 45% de los monogramas en un texto en castellano son vocales. Y no solo eso, sino que además de monogramas también debemos buscar la redundancia de digramas, trigramas y poligramas comunes. Por ejemplo, digramas comunes en castellano son “de”, “en”, “la”, “el”, etc.

3.3 Códigos de carácter general para el cifrado y descifrado

Por todo lo anterior, para el desarrollo de este trabajo se requiere de un código que realice la asociación de letras y números, así como de números a letras y otro que cuente la cantidad de veces que se repiten los monogramas.

El siguiente programa, cuyo código aparece en el Anexo III: Códigos denominado como A III: Código 1 Asociación 1, realiza la primera asociación, de letras a números, y a continuación “traduce” un texto cualquiera en mayúsculas y sin espacios. Para probar su eficacia se han utilizado frases pertenecientes a la carta que Eisenhower escribió a las tropas aliadas que iban a participar en la “Operación Overlord” (día D). [23]

A III: Código 1 Asociación 1

Texto Original: ESTAISAPUNTODEEMBARCAROSENLAGRANCRUZADALOSOJOSDELMUNDOOSCONT
EMPLAN

Equivalente numérico: [4, 18, 19, 0, 8, 18, 0, 15, 20, 13, 19, 14, 3, 4, 4, 12, 1, 0, 17, 2, 0, 17, 14, 18, 4, 13, 11, 0, 6, 17, 0, 13, 2, 17, 20, 25, 0, 3, 0, 11, 14, 18, 14, 9, 14, 18, 3, 4, 11, 12, 20, 13, 3, 14, 14, 18, 2, 14, 13, 19, 4, 12, 15, 11, 0, 13]

Por otro lado, para realizar la acción contraria se utilizaría el siguiente código (A III: Código 2 Asociación inversa 1). El texto asociado es un fragmento del discurso que W. Churchill pronunció en la Cámara de los Comunes del Parlamento británico en 1940 tras llevar a cabo la “Operación Dinamo”, también conocida como el Milagro de Dunkerque. [24]

A III: Código 2 Asociación inversa 1

Cadena de números: [11, 20, 2, 7, 0, 17, 4, 12, 14, 18, 4, 13, 11, 0, 18, 15, 11, 0, 24, 0, 18, 11, 20, 2, 7, 0, 17, 4, 12, 14, 18, 4, 13, 11, 0, 18, 15, 8, 18, 19, 0, 18, 3, 4, 0, 19, 4, 17, 17, 8, 25, 0, 9, 4, 11, 20, 2, 7, 0, 17, 4, 12, 14, 18, 4, 13, 11, 14, 18, 2, 0, 12, 15, 14, 18, 24, 4, 13, 11, 0, 18, 2, 0, 11, 11, 4, 18, 11, 20, 2, 7, 0, 17, 4, 12, 14, 18, 4, 13, 11, 0, 18, 2, 14, 11, 8, 13, 0, 18, 13, 20, 13, 2, 0, 13, 14, 18, 17, 4, 13, 3, 8, 17, 4, 12, 14, 18]

texto asociado: LUCHAREMOSENLASPLAYASLUCHAREMOSENLASPISTASDEATERRIZAJELUCHAR
EMOSENLOSCAMPOSYENLASCALLESLUCHAREMOSENLASCOLINASNUNCANOSRENDIREMOS

Por último, se adjunta un código (A III: Código 3 Contador de letras) que cuenta las veces que se repite cada letra en el texto y entrega dicho valor. Para ver su eficacia se ha utilizado un fragmento conocido de la obra *La vida es sueño* de Calderón de la Barca, cortesano, poeta, soldado y sacerdote español, autor de los conocidos versos “Aquí la más principal hazaña es obedecer...” que aparecen en la obra *Para vencer a amor querer vencerle*. El mensaje sería el siguiente: "Apurar, cielos, pretendo ya que me tratáis así, qué delito cometí contra vosotros naciendo; aunque si nací, ya entiendo qué delito he cometido. Bastante causa ha tenido vuestra justicia y rigor; pues el delito mayor del hombre es haber nacido". Esta función sirve para cualquier tipo de texto, no necesariamente en mayúsculas y sin espacios. La frecuencia que devuelve la función es la siguiente:

A III: Código 3 Contador de letras

A:20
B:3
C:9
D:10
E:24
F:0
G:1
H:4
I:14
J:1
K:0
L:6
M:5
N:11
O:19
P:3
Q:4
R:12
S:12
T:16
U:10
V:2
W:0
X:0
Y:4
Z:0

Como puede observarse las letras con mayor frecuencia son en este orden: E (24), A (20), O (19), S (16), I (14), etc. En la gráfica se aprecia como la frecuencia de uso de los monogramas del alfabeto se mantiene aproximadamente constante.

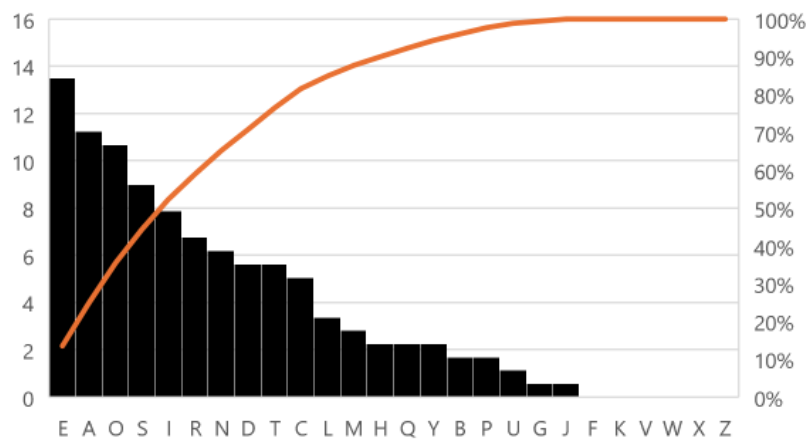


Figura 3-3 Frecuencia de uso de cada monograma (autoría propia)

Antes de comenzar con los fundamentos matemáticos propiamente dicho se establece una nomenclatura común para mencionar ciertos elementos.

- El mensaje en claro viene representado por m .
- La clave utilizada viene dada por k .
- La función encriptadora del sistema F , normalmente se representará como Función (mensaje, clave), esto es, $F(k,m)$.
- El mensaje encriptado m' , de tal forma que $F(k,m) = m'$.
- Una función de desencriptado G , donde $G(k,m') = m$.

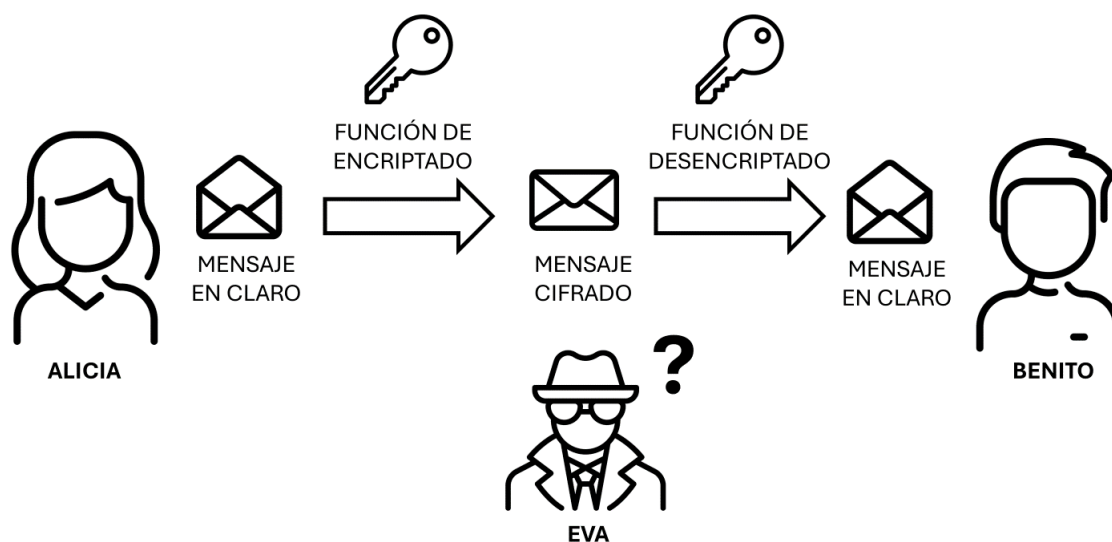


Figura 3-4 Proceso de encriptado y desencriptado de mensajes (autoría propia)

3.4 Números primos y Divisibilidad

El conjunto de los números enteros se denota como $\mathbb{Z}$. Los enteros pueden sumarse, restarse y multiplicarse entre ellos obteniendo como resultado otro número entero, sin embargo, no siempre puede realizarse una división entre ellos dando lugar a un entero. Esta característica se llama divisibilidad. Para $a, b \in \mathbb{Z}$ se dice que a divide b , escrito como $a|b$ si existe $c \in \mathbb{Z}$ tal que $a \cdot c = b$. En caso contrario se escribe $a \nmid b$.

Si $a|b$ y a es positivo, entonces a es un divisor de b . Además, si $a \notin \{1, b\}$, entonces a es factor de b . Un número positivo $p > 1$ es primo si no tiene factores, es decir, si sus únicos divisores son el número 1 y él mismo. Un número entero positivo mayor que 1 que no es primo recibe el nombre de compuesto. Por convención, el número 1 no es ni primo ni compuesto.

Todo número entero mayor que 1 puede expresarse de manera única como producto de primos. Por tanto, cualquier entero positivo $N > 1$ se puede escribir como $N = \prod_i p_i^{e_i}$, donde p_i son primos distintos y $e_i \geq 1$ para todo i . Aunque puede parecer una cuestión sencilla, el fundamento detrás de esta expresión es en realidad bastante complejo.

3.5 Máximo común divisor

El máximo común divisor de dos enteros a, b se escribe como $MCD(a,b)$ y es el número entero más grande c tal que $c|a$ y $c|b$. Por ejemplo, el MCD de 27 y 15 es 3, ya que este es el mayor de los divisores comunes de ambos números. Por tanto, $MCD(27,15)=3$. Para un número primo p se tiene que $MCD(a,p)$ es igual a 1 o a p . Si $MCD(a,b)=1$ entonces a y b son primos relativos o coprimos.

Además dados dos números positivos a, b , entonces existen dos enteros u y v tales que $ua + vb = \text{MCD}(a, b)$. Además, $\text{MCD}(a, b)$ es el entero positivo más pequeño que pueda ser expresado de esa manera.

3.6 Aritmética Modular

Dados $a, b, N \in \mathbb{Z}$ con $N > 1$. Se utiliza la notación $[a \bmod N]$ para denotar el resto de a al ser dividido por N . Para un número entero a y un número positivo b , existen enteros únicos q, r tales que $a = p \cdot q + r$ y $0 \leq r < b$. Se define $0 \leq r < b$ como igual a r y $0 \leq [a \bmod N] < b$. Por ejemplo 2 y 5 son congruentes en módulo 3 ya que $5 = 3 \cdot 1 + 2$ y se escribe $2 \equiv 5 \bmod 3$.

Se dice que a y b son congruentes en módulo N , si el resto de dividir a por N es el mismo que el de dividir b por N . Esto se escribe $a \equiv b \bmod N$, es decir, $[a \bmod N] = [b \bmod N]$. Esto se cumple si y solo si $N|(a - b)$.

La congruencia en módulo N es reflexiva ($a = a \bmod N$ para todo a), simétrica ($a = b \bmod N$ implica que $b = a \bmod N$) y transitiva (si $a = b \bmod N$ y $b = c \bmod N$, entonces $a = c \bmod N$).

Por otro lado debe introducirse la definición de inverso modular. Si para un entero b existe un entero c tal que $b \cdot c = 1 \bmod N$, se dice que b es invertible módulo N y llamamos a c un inverso multiplicativo de b módulo N .

Finalmente, dados b, N enteros, con $b \geq 1$ y $N \geq 1$. Entonces b es invertible módulo N si y solo si $\text{MCD}(b, N) = 1$. Esto se debe a que si existe $bc = 1 \bmod N$, esto implica que el resto de dividir bc entre N es igual a 1 y por tanto existe un número d tal que $bc = dN + 1 \Rightarrow bc - dN = 1$. Como el $\text{MCD}(b, N)$ es el número entero positivo más pequeño que pueda expresarse como $ub + vN$ (3.5 Máximo común divisor) y el entero positivo más pequeño es el 1, entonces $\text{MCD}(b, N) = 1$.

Los enteros X e Y que satisfacen $ub + vN$ pueden encontrarse utilizando el Algoritmo de Euclides extendido como se verá posteriormente.

3.7 Algoritmo de Euclides

El algoritmo de Euclides permite calcular el máximo común divisor de dos números enteros. La base de este algoritmo es el hecho de que el máximo común divisor de dos números enteros no varía si al mayor de ambos se le resta el otro número. Esta operación puede repetirse hasta que los números tienen el mismo valor, en ese caso, se ha obtenido el máximo común divisor.

El Algoritmo de Euclides encuentra el máximo común divisor cuando uno alguno de los números a, b es cero. Esto se debe a que $\text{MCD}(a, 0) = a$ y $\text{MCD}(b, 0) = b$, ya que el mayor entero que puede dividir tanto a como 0 es a . Lo mismo ocurriría para b .

Por otro lado para realizar la reducción de los números a y b se debe demostrar que $\text{MCD}(a, b) = \text{MCD}(b, r) = \text{MCD}(b, a - b)$, donde $a = b \cdot q + r$. El $\text{MCD}(a, b)$ es divisor tanto de b como de $a - b$, por tanto, $\text{MCD}(a, b)$ es un divisor común de b y $a - b$. Como $\text{MCD}(b, a - b)$ es el mayor de los divisores comunes de b y $a - b$, entonces se cumple que $\text{MCD}(a, b) \leq \text{MCD}(b, a - b)$. Lo mismo ocurre para b , de donde se deduce que $\text{MCD}(b, a - b) \leq \text{MCD}(a, b)$.

Considerando todo lo anterior, si $\text{MCD}(a, b) \leq \text{MCD}(b, a - b)$ y $\text{MCD}(b, a - b) \leq \text{MCD}(a, b)$, entonces, $\text{MCD}(b, a - b) = \text{MCD}(a, b)$. Si se aplica de forma repetida esta proposición se tiene que:

$$\text{MCD}(a, b) = \text{MCD}(b, a - b) = \text{MCD}(b, a - b - b) = \text{MCD}(b, a - b - b - b \dots) = \text{MCD}(b, a - qb)$$

Como $a = bq + r$, entonces, $r = a - qb$ y se tiene finalmente que $\text{MCD}(a, b) = \text{MCD}(b, r)$.

A continuación, aparecen las iteraciones para hallar el máximo común divisor de 195 y 52.

195	-	52	→	143
143	-	52	→	91
91	-	52	→	39
52	-	39	→	13
39	-	13	→	26
26	-	13	→	13
13	-	13	→	0

Figura 3-5 Iteraciones para calcular el MCD de 195 y 52 (autoría propia)

Por tanto se obtiene que $MCD(195,52)=13$. El código A III: Código 4 Algoritmo de Euclides obtiene el máximo común divisor de dos números utilizando el algoritmo de Euclides.

A III: Código 4 Algoritmo de Euclides
 $MCD(195, 52) : 13$

3.8 Algoritmo de Euclides extendido

El algoritmo de Euclides extendido permite, además de encontrar el máximo común divisor de dos números, expresar este como una combinación lineal de ambos. De tal forma que dados dos números a y b , además de calcular su máximo común divisor se tiene que $MCD(a,b) = au + bv$. Para ello, en cada iteración se debe expresar a como el producto de b por un número entero, más el resto. Por ejemplo, para los números $a = 78$ y $b = 32$, debe expresarse a de esta forma:

$$a = b \cdot q + r$$

$$78 = 32 \cdot 2 + 14$$

En cada iteración a tomará el valor de b , y b el del resto de la división, hasta que el valor del resto sea nulo.

$$78 = 32 \cdot 2 + 14$$

$$32 = 14 \cdot 2 + 4$$

$$14 = 4 \cdot 3 + 2$$

$$4 = 2 \cdot 2 + 0$$

Por tanto el $MCD(78,32) = 2$. A continuación, se despeja el resto de cada iteración y se expresa de la forma $r = a - b \cdot q$ y se sustituye su valor en la expresión anterior.

$$a = b \cdot q + r \Rightarrow r = a - b \cdot q$$

$$78 = 32 \cdot 2 + 14 \Rightarrow 14 = 78 - 32 \cdot 2 \quad (3)$$

$$32 = 14 \cdot 2 + 4 \Rightarrow 4 = 32 - 14 \cdot 2 \quad (2)$$

$$14 = 4 \cdot 3 + 2 \Rightarrow 2 = 14 - 4 \cdot 3 \quad (1)$$

$$4 = 2 \cdot 2 + 0$$

$$(1) \ 2 = 14 - 4 \cdot 3 = (2) \ 14 - (32 - 14 \cdot 2) \cdot 3 = 14 - 32 \cdot 3 + 6 \cdot 14 = -3 \cdot 32 + 7 \cdot 14 = (3) \ -3 \cdot 32 + 7 \cdot (78 - 32 \cdot 2) = -3 \cdot 32 + 7 \cdot 78 - 14 \cdot 32 = 78 \cdot 7 - 17 \cdot 32 \Rightarrow 2 = 78 \cdot 7 - 32 \cdot 17$$

Por ello los valores de u y v son respectivamente 7 y -17. El siguiente código (A III: Código 5 Algoritmo de Euclides Extendido) aplica el algoritmo extendido de Euclides para obtener el MCD y los valores u y v .

A III: Código 5 Algoritmo de Euclides Extendido

MCD: 2 u: 7 v: -17

Una de las aplicaciones útiles de este algoritmo es el cálculo de inversos modulares. El inverso modular solo existe si $MCD(a,b)=1$. Para dos números coprimos entre ellos, por ejemplo 15 y 8, al ser 1 su máximo común divisor, se sabe que el inverso de 15 en módulo 8 existe, además se tiene que $1 = 15 \cdot (-1) + 8 \cdot 2$ y por tanto, $u = -1$ y $v = 2$. De lo anterior se deduce que el inverso modular de 15 en módulo 8 es 7, ya que $-1 \bmod 8 = 7$.

3.9 Función ϕ de Euler

La función de Euler establece el número de enteros positivos menores o iguales que un número n que son coprimos con él. Es decir, indica que cantidad de números entre 1 y n no comparten ningún divisor común con n . Se denota como $\phi(n)$. Por ejemplo si $n=4$, se cumple que $\phi(4)=2$ ya que 1 y 3 son primos relativos de 4. Para todo n se cumple que $\phi(n) \leq n-1$.

Por otro lado si n es un número primo, entonces se cumple que todo entero positivo menor que n es coprimo con él. Por lo tanto para todo número primo p se cumple que $\phi(p)=p-1$.

3.10 Teorema de Euler

El Teorema de Euler define que para un número n entero positivo y un número a coprimo con n se tiene que $a^{\phi(n)} \equiv 1 \pmod{n}$. El Pequeño Teorema de Fermat, que se describe más adelante y se emplea en el algoritmo RSA, es un caso particular de este teorema, en el cual p es un número primo y por tanto, se cumple que $\phi(p)=p-1$ y por tanto, $a^{p-1} \equiv 1 \pmod{p}$.

3.11 Teorema Chino de los Restos

El Teorema Chino de los Restos es fundamental para el correcto funcionamiento del algoritmo RSA. El teorema afirma que siendo $n_1, n_2, \dots, n_k$ números primos distintos, enteros positivos, existe un número entero x , que resuelve el sistema congruente:

$$x \equiv a_1 \pmod{n_1}$$

$$x \equiv a_2 \pmod{n_2}$$

...

$$x \equiv a_k \pmod{n_k}$$

Y además, todas las soluciones de dicho sistema son congruentes módulo el producto $N = n_1 \cdot n_2 \cdot \dots \cdot n_k$. En concreto, en este caso, siendo dos números de dichas condiciones p y q tales que $N = p \cdot q$, el siguiente sistema tiene solución:

$$x \equiv a_1 \pmod{p}$$

$$x \equiv a_2 \pmod{q}$$

Además dicha solución es única en módulo N . Por otro lado, existe otra propiedad, que utilizada con el teorema chino de los restos, permite ganar eficiencia aritmética y facilita el cálculo. Dicha propiedad es $a \equiv b \pmod{n} \Rightarrow a \equiv b \pmod{d}$, siendo d divisor del número n . Por tanto, si unificamos estas propiedades, para los números mencionados anteriormente $N = p \cdot q$, se tiene que:

$$a \equiv b \pmod{N} \Leftrightarrow a \equiv b \pmod{p} \text{ y } a \equiv b \pmod{q}$$

Por tanto, $M = C^d \cdot \bmod N$, que es la operación que se realiza en cifrado de RSA puede simplificarse en dos operaciones:

$$M \equiv C^d \pmod{p}$$

$$M \equiv C^d \pmod{q}$$

A raíz de esto, el calculo será más rápido ya que el tamaño de p y q es mucho menor que el de N - del orden de la mitad- y además pueden calcularse ambas operaciones al mismo tiempo. Además, el Teorema Chino de los Restos se utiliza con otro teorema para generar las claves. Este teorema es el Pequeño Teorema de Fermat.

3.12 Pequeño Teorema de Fermat

Este teorema justifica que dado un número primo p , y un número a coprimo de p , entonces se cumple que:

$$a^{p-1} \equiv 1 \pmod{p}$$

Por tanto, puede deducirse que si $\phi(p) \equiv p-1$, entonces:

$$a^{\phi(p)} \equiv 1 \pmod{p}$$

Lo mismo ocurriría para q . Por todo ello, cuando se realiza el descryptado:

$$C^d \pmod{N} = (M^e)^d \pmod{N} = M^{ed} \pmod{N}$$

Y dado que $ed \equiv 1 \pmod{\phi(N)}$, entonces:

$$\begin{aligned} M^{ed} \pmod{N} &= M^{\phi(N)K + 1} \pmod{N} = M^{\phi(N)K} \cdot M^1 \pmod{N} = \\ &= 1^K \cdot M \pmod{N} = M \pmod{N} \end{aligned}$$

4 CRIPTOGRAFÍA DE CLAVE SIMÉTRICA

4.1 Cifra del César

Uno de los sistemas de cifrado más antiguos es el cifrador del César o cifra del César. Recibe su nombre en honor al dictador romano Julio César, quien teóricamente lo utilizaba para enviar mensajes secretos durante sus campañas.

Este sistema de encriptado es un sistema por sustitución monoalfabético, es decir, cada letra del texto original a encriptar es sustituida por otra letra que se encuentra a un número k de posiciones más adelante en el mismo alfabeto. Este tipo de cifradores también se conocen como cifradores monoalfabéticos por desplazamiento puro. En el caso del cifrado del César, el valor que toma k es 3, de tal forma que el alfabeto obtenido 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 4-1 Alfabeto cifrado de Julio César (autoría propia)

Este sistema de cifrado, aunque adecuado para la época, en realidad es bastante sencillo y presenta un nivel de seguridad muy bajo como se demostrará en un apartado posterior al someterlo a un criptoanálisis.

Matemáticamente, se puede decir que en este sistema un mensaje m , compuesto por una sucesión de caracteres $m_1, m_2, m_3 \dots$, donde cada m_i es un número entero, que se corresponde por una letra, se suma a un valor $k \in N$, que, en este caso, sería la clave. Por tanto, la función de encriptado se define como:

$$F(m_i, k) = m_i + k \mod 26$$

En el caso del descifrado, la tarea a realizar es la inversa, a cada m_i se le resta el valor de la clave k . La función de descifrado es de la forma:

$$G(m_i', k) = m_i' - k \mod 26$$

En el siguiente código se emplea el algoritmo del César para cifrar el siguiente mensaje: “*Alea iacta est*”, una frase atribuida al personaje que da nombre a este sistema de cifrado y significa “La suerte está echada”.

En primer lugar, calculamos el equivalente numérico con el código (A III: Código 1 Asociación 1) que se explicó anteriormente:

A III: Código 1 Asociación 1

Texto Original: ALEAIAC TAEST

Equivalente numérico: [0, 11, 4, 0, 8, 0, 2, 19, 0, 4, 18, 19]

A continuación, es cifrado utilizando la cifra del César. Esta función (A III: Código 6 Cifra del César) se encarga de sumar a cada número de la lista el valor k , al que se ha llamado desplazamiento. En este caso $k=3$.

A III: Código 6 Cifra del César

Cadena inicial: [0, 11, 4, 0, 8, 0, 2, 19, 0, 4, 18, 19]

Cadena encriptada: [3, 14, 7, 3, 11, 3, 5, 22, 3, 7, 21, 22]

Por último se ejecuta A.2 Asociación 1 inversa (A III: Código 2 Asociación inversa 1) para obtener el mensaje cifrado.

A III: Código 2 Asociación inversa 1

Cadena de números: [3, 14, 7, 3, 11, 3, 5, 22, 3, 7, 21, 22]

texto asociado: DOHDLDFWDHVW

Cuando el destinatario recibiera el mensaje, debería restar al valor de cada letra el número 3, tal y como se observa en el siguiente fragmento de código (A III: Código 7 Descifrado César), que realiza el descifrado.

A III: Código 7 Descifrado César

Cadena_encriptada: [3, 14, 7, 3, 11, 3, 5, 22, 3, 7, 21, 22]

Cadena original: [0, 11, 4, 0, 8, 0, 2, 19, 0, 4, 18, 19]

Finalmente, se realiza el criptoanálisis de dicho texto. Como se mencionó anteriormente, este sistema de cifrado no es muy robusto, en este caso solo hay 26 claves posibles, una para cada letra, y en general, tantas como letras tenga el abecedario que utilizemos. Por ello, la primera forma de romper el cifrado es por “fuerza bruta”, esto es, probar todas las claves posibles. Para realizar esta acción utilizamos el siguiente código (A III: Código 8 Criptoanálisis César por fuerza bruta), y posteriormente realizamos la asociación inversa para obtener todos los posibles resultados.

A III: Código 8 Criptoanálisis César por fuerza bruta**A III: Código 2 Asociación inversa 1**

Cadenas de numeros: [[3, 14, 7, 3, 11, 3, 5, 22, 3, 7, 21, 22], [4, 15, 8, 4, 12, 4, 6, 23, 4, 8, 22, 23], [5, 16, 9, 5, 13, 5, 7, 24, 5, 9, 23, 24], [6, 17, 10, 6, 14, 6, 8, 25, 6, 10, 24, 25], [7, 18, 11, 7, 15, 7, 9, 0, 7, 11, 25, 0], [8, 19, 12, 8, 16, 8, 10, 1, 8, 12, 0, 1], [9, 20, 13, 9, 17, 9, 1, 2, 9, 13, 1, 2], [10, 21, 14, 10, 18, 10, 12, 3, 10, 14, 2, 3], [11, 22, 15, 11, 19, 11, 13, 4, 11, 15, 3, 4], [12, 23, 16, 12, 20, 12, 14, 5, 12, 16, 4, 5], [13, 24, 17, 13, 21, 13, 15, 6, 13, 17, 5, 6], [14, 25, 18, 14, 22, 14, 16, 7, 14, 18, 6, 7], [15, 0, 19, 15, 23, 15, 17, 8, 15, 19, 7, 8], [16, 1, 20, 16, 24, 16, 18, 9, 16, 20, 8, 9], [17, 2, 21, 17, 25, 17, 19, 10, 17, 21, 9, 10], [18, 3, 22, 18, 0, 18, 20, 11, 18, 22, 10, 11], [19, 4, 23, 19, 1, 19, 21, 12, 19, 23, 11, 12], [20, 5, 24, 20, 2, 20, 22, 13, 20, 24, 12, 13], [21, 6, 25, 21, 3, 21, 23, 14, 21, 25, 13, 14], [22, 7, 0, 22, 4, 22, 24, 15, 22, 0, 14, 15], [23, 8, 1, 23, 5, 23, 25, 16, 23, 1, 15, 16], [24, 9, 2, 24, 6, 24, 0, 17, 24, 2, 16, 17], [25, 10, 3, 25, 7, 25, 1, 18, 25, 3, 17, 18], [0, 11, 4, 0, 8, 0, 2, 19, 0, 4, 18, 19], [1, 12, 5, 1, 9, 1, 3, 20, 1, 5, 19, 20], [2, 13, 6, 2, 10, 2, 4, 21, 2, 6, 20, 21]]

textos asociados: ['DOHDLDFWDHVW', 'EPIEMEGXEIWX', 'FQJFNHXYFJXY', 'GRKGOGIZGKYZ', 'HSLHPHJAHZA', 'ITMIQIKBIMAB', 'JUNJRJLCJNBC', 'KVOKSKMDKOC', 'LWPLTLNELPDE', 'MXQMUMOFMQEF', 'NYRNVNPGNRFG', 'OZSOWOQHOSGH', 'PATXPRIPTHI', 'QBUQYQSJQUIJ', 'RCVRZRTKRVJK', 'SDWSASULSWKL', 'TEXTBTVMTXLM', 'UFYUCUWNUYMN', 'VGZVDVXOVZNO', 'WHAWEWYPWAOP', 'XIBXFXZQXBPQ', 'YJCYGYARYCQR', 'ZKDZHZBSZDRS', '**ALEAIAC TAEST**', 'BMFBJBDUBFTU', 'CNGCKCEVCGUV']

4.2 Cifra de Vigenère

El cifrado de Vigenère, que debe su nombre al criptólogo francés Blaise de Vigenère, es un sistema de cifrado por sustitución polialfabética. Esto significa que cada letra del mensaje a cifrar puede corresponderse con letras distintas del mensaje cifrado. Por ejemplo, la letra “A” puede corresponderse con la letra “D” la primera vez que aparece y con la letra “I” la segunda vez.

Este método, similar al del César, por tanto, un desplazamiento de cada carácter del mensaje en módulo 26. La diferencia radica en que el desplazamiento no es un número de valor constante, sino que depende del valor numérico asociado a los caracteres de una palabra clave, que se repite de forma cíclica y se escribe bajo el mensaje, como se observa en el ejemplo.

M E N S A J E O R I G I N A L
C L A V E C L A V E C L A V E
O P N O E L H O M M I T N V P

Figura 4-2 Ejemplo de cifrado utilizando el sistema de Vigenère (autoría propia)

Para realizar este cifrado, es útil utilizar la tabla de Vigenère.

	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	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	A
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	B
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	C
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	D
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	E
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	F
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	G
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	H
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	I
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	J
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	K
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	L
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	M
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	N
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	O
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	P
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	Q
S	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	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	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	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	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	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	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	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

Figura 4-3 Tabla de Vigenère [9]

Su descripción matemática sería la siguiente.

$$F(m_i, k_i) = m_i + k_i \mod 26$$

Donde $m = m_1, m_2, m_3 \dots$ es el mensaje y $k = k_1, k_2, k_3 \dots$ es la clave, de tal forma que cada m_i y k_i corresponden con el número entero equivalente de la letra que ocupa esa posición i . Para descifrarlo habría que realizar la operación contraria, restar k_i , de tal manera que la función de descifrado es de la forma:

$$G(m'_i, k_i) = m'_i - k_i \mod 26$$

A continuación, se cifra la siguiente frase: “Todo arte de la guerra se basa en el engaño” que aparece en *El arte de la guerra* del general chino Sun Tzu (722-481 a. C). La clave utilizada será “SUNTZU”. Para ello utilizaremos el siguiente código (A III: Código 9 Cifra de Vigenère).

A III: Código 1 Asociación 1

Mensaje:

Texto Original: TODOARTEDELAGUERRASEBASAENELENGANO

Equivalente numérico: [19, 14, 3, 14, 0, 17, 19, 4, 3, 4, 11, 0, 6, 20, 4, 17, 0, 18, 4, 1, 0, 18, 0, 4, 13, 4, 11, 4, 13, 6, 0, 13, 14]

Clave:

Texto Original: SUNTZU

Equivalente numérico: [18, 20, 13, 19, 25, 20]

A III: Código 9 Cifra de Vigenère**A III: Código 2 Asociación inversa 1**

mensaje original: [19, 14, 3, 14, 0, 17, 19, 4, 3, 4, 11, 0, 6, 20, 4, 17, 1, 7, 0, 18, 4, 1, 0, 18, 0, 4, 13, 4, 11, 4, 13, 6, 0, 13, 14]

mensaje cifrado: [11, 8, 16, 7, 25, 11, 11, 24, 16, 23, 10, 20, 24, 14, 17, 10, 16, 20, 10, 24, 14, 19, 17, 20, 22, 7, 17, 4, 3, 7, 24, 20, 0, 7]

texto asociado: LIQHZLLYQXKUYORKQUKYOTRUWHREDHYUAH

Para descifrarlo, se debe restar el valor de la clave al texto resultante de forma cíclica. Para ello se utiliza la misma función de descifrado, pero utilizando previamente otra función (A III: Código 10 Descifrado Vigenère) que hace negativo el valor de los números, de tal forma que al realizar la suma se resten.

A III: Código 10 Descifrado Vigenère**A III: Código 2 Asociación inversa 1**

mensaje cifrado: [11, 8, 16, 7, 25, 11, 11, 24, 16, 23, 10, 20, 24, 14, 17, 10, 16, 20, 10, 24, 14, 19, 17, 20, 22, 7, 17, 4, 3, 7, 24, 20, 0, 7]

mensaje descifrado: [19, 14, 3, 14, 0, 17, 19, 4, 3, 4, 11, 0, 6, 20, 4, 17, 17, 0, 18, 4, 1, 0, 18, 0, 4, 13, 4, 11, 4, 13, 6, 0, 13, 14]

texto asociado: TODOARTEDELAGUERRASEBASAENELENGANO

De forma esquemática la operación de descifrado quedaría así:

```

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

```

Figura 4-4 Descifrado del mensaje restando la clave SUNTZU (autoría propia)

A continuación, se procede con el criptoanálisis del texto. Para ello debe utilizarse el método de Kasiski. El primer paso para aplicar este método consiste en averiguar la longitud de la clave, k , para ello, se utiliza el algoritmo de Euclides, explicado con anterioridad.

Supongamos que se intercepta el siguiente mensaje y se presupone que ha sido cifrado utilizando el sistema de Vigenère.

Mensaje: “YFAGNLORBJEDDYNBYLHMISHJPEGJHBYJHDBYIRSDYRNZOEFNXIERUT
BMIDRJFGBMINDDYNBUFEIJBURBISFXFOYXMEFYJBBXENIUBNLBEQNMUFXDFOBIN
QDLOFLOAYMISRBWAEJVAWXMDRLLIFCULANARB”.

Se debe buscar en el texto la repetición de monogramas, digramas, tigramas, etc.. Es decir, conjuntos de letras que se repiten. En este caso:

Mensaje: “YFAGNLORBJEDD**YN**BYLH**MIS**HJPEGJHBYJHDBYIRSD**YRN**ZOEFNXIERUT
B**MID**RJFGB**MI**ND**DYN**BUFEIJBURBISFXFOYXMEFYJBBXENIUBNLBEQNMUFXDFOBIN
QDLOFLOAY**MI**SRBWAEJVAWXMDRLLIFCULANARB”.

Puede observarse que los digramas “DY” y “MI” se repiten con bastante frecuencia. Ahora se deben contar los caracteres que separan dichos digramas. Se obtiene que para “DY” hay 28 caracteres entre los digramas, y para “MI” hay 36, 8 y 64 caracteres. A continuación, se calcula el máximo común divisor de 28 con cualquiera de los tres números, por ejemplo, el 36, utilizando la función del algoritmo de Euclides (A III: Código 4 Algoritmo de Euclides).

A III: Código 4 Algoritmo de Euclides

MCD (28, 36) : 4

Podría ser casualidad, pero si se realiza esta operación con el resto de valores, en efecto, se obtiene que el máximo común divisor de esos cuatro números es 4. Por tanto, sospechamos que la clave tiene cuatro caracteres. Esto ocurre ya que al haber poligramas con mayor tendencia a aparecer en el texto y que se repiten con mayor frecuencia es bastante probable que en algún momento coincidan con las mismas letras de la clave, y se cifren de la misma manera.

Considerando todo lo anterior, ahora el criptoanalista se encontraría ante cuatro textos distintos, cifrados utilizando un sistema de sustitución monoalfabético, es decir, la cifra del César. Dichos textos son el resultado de tomar los caracteres que ocupan las posiciones 1, 2, 3 y 4 de cada grupo de cuatro letras.

1 2 3 4 1 2 3 4 1 2 3 4 1 2 3 4 1 2 3 4 1 2 3 4 1 ...
Y F A G N L O R B J E D D Y N B Y Y L H M I S H J

Figura 4-5 Obtención de los textos relativos a los caracteres de la clave (autoría propia)

Por ejemplo, tomando los caracteres que ocupan la segunda posición se obtiene el siguiente mensaje, al que se ha denominado Mb.

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

Mb: “FLJYYIPHHIYOXUIFIYFBIFMYXUBMDILOIWVMLUA”.

Utilizando el siguiente código (A III: Código 11 Método Kasiski), se obtienen los textos que contienen los caracteres que ocupan las posiciones relativas 1, 2, 3 y 4. Dichos textos son Ma, Mb, Mc y Me respectivamente.

A III: Código 11 Método Kasiski

Ma: YNBDYMJJYDZNRMJMDUJBXXYBILNXBDLMBJXLCN, Mb: FLJYYIPHHIYOXUIFIYFBIFMYXUBMDILOIWVMLUA, Mc: AOENLSEBDRREITDGNNEUSOEJEBEUONOASAADILR, Md: , Me: GRDBHHGYBSNFEBRDBIRFYFBNNQFFQFYREWRFAB

Llegados a este punto, el criptoanalista debe utilizar las técnicas de redundancia del lenguaje planteadas en la introducción. Como se mencionó en las consideraciones, las letras más repetidas estadísticamente en castellano son E, A, S y O. Esto permite aplicar un Método por Coincidencia múltiple, denominado Regla EAOS. Esta regla, se basa en buscar un grupo de cuatro letras en cada uno de los textos, con una redundancia considerable en el mensaje, cuyas posiciones relativas sean las mismas que las letras AEOS en el alfabeto. Es decir, entre la A y la E hay cuatro espacios, diez entre la E y la O y la O y la S distan cuatro caracteres. En otras palabras, lo que se busca es encontrar las letras cifradas que se corresponden con estas cuatro letras, para ello se deben seleccionar las cuatro letras que más se repiten, sin embargo, como es posible que alguna de estas letras no sean E, A, O o S realmente, para asegurar se busca que la distancia entre ellas se corresponda con la distancia entre esas cuatro letras. En la siguiente figura aparece en la primera fila cada letra del alfabeto, y en las siguientes el número de veces que se repite cada letra para los textos Ma, Mb, Mc y Me. De esta distribución se deduce que letra

se corresponde con la “A” en el alfabeto en claro y cual ha sido el desplazamiento de cada submensaje. Para nuestros respectivos textos, podríamos intuir la siguiente clave, teniendo en cuenta que se identifica la A con la casilla que se encuentra tachada.

	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	0	5	1	4	0	0	0	0	1	6	0	3	4	4	0	0	0	1	0	0	1	0	0	4	4	1
b	1	2	0	1	0	4	0	2	7	1	0	3	3	0	2	1	0	0	0	0	3	1	1	2	5	0
c	4	2	0	3	7	0	1	0	2	1	0	2	0	4	4	0	0	3	3	1	2	0	0	0	0	0
e	1	7	0	2	2	7	2	2	1	0	0	0	0	3	0	0	2	5	1	0	0	0	1	0	3	0

Figura 4-6 Regla EAOS aplicada al texto que se pretende criptoanalizar (autoría propia)

Por tanto, la clave usada en cada uno de los mensajes sería “J” para Ma, “U” para Mb, “A” para Mc y “N” para Me y el desplazamiento 9, 20, 0 y 13 respectivamente. Por tanto, la clave que cifra el mensaje sería K: JUAN. Si utilizamos dicha clave para descifrar el texto con el código (A III: Código 10 Descifrado Vigenère) explicado anteriormente se obtiene.

A III: Código 10 Descifrado Vigenère

A III: Código 2 Asociación inversa 1

texto asociado: PLATEROESPEQUENOPELUDOSUAVETANBLANDOPORFUERAQUESEDIRIATODODE
ALGODONQUENOLLEVAHUESOSSOLOLOSESEPEJOSDEAZABACHEDESUSOJOSSONDUROSCUALDOSESCAR
ABAJOSDECRISTALNEGRO

El mensaje obtenido tiene sentido en castellano, “Platero es pequeño, peludo, suave; tan blando por fuera, que se diría todo de algodón, que no lleva huesos. Sólo los espejos de azabache de sus ojos son duros cual dos escarabajos de cristal negro.” Se trata del primer párrafo de la obra *Platero y yo*, de Juan Ramón Jiménez.

4.3 Criptosistema de Hill

En 1929, Lester S. Hill publica un artículo en *The American Mathematical Monthly*, en el cual plantea el uso de la multiplicación matricial para el cifrado de mensajes. [17]

En este sistema, el mensaje original se divide en grupos de caracteres de tamaño fijo, cuya longitud viene determinada por el orden de la matriz clave, y se realiza el encriptado independiente de cada bloque.

Es un sistema polialfabético, ya que cada letra del mensaje original puede corresponder con más de una letra del mensaje cifrado, y además de trata de un cifrador poligrámico ya que, en lugar de cifrar carácter a carácter, necesita un número $n > 1$, para componer un bloque de cifrado, dando lugar al cifrador de digramas, trigramas, o en general, poligramas.

Para un mensaje original $m = m_1, m_2, m_3 \dots$ que desea cifrarse con el sistema de Hill, con una matriz clave $n \times n$, se divide el texto original en vectores de longitud n , y se multiplica la matriz clave por cada vector. Por ejemplo, para una matriz clave 2×2 , el cifrado de cada uno de los elementos resultantes m_i' sería el siguiente.

$$m_1' = (k_{11} \cdot m_1 + k_{12} \cdot m_2) \text{ módulo } 26$$

$$m_2' = (k_{21} \cdot m_1 + k_{22} \cdot m_2) \text{ módulo } 26$$

$$\begin{pmatrix} m_1' \\ m_2' \end{pmatrix} = \begin{pmatrix} k_{11} & k_{12} \\ k_{21} & k_{22} \end{pmatrix} \begin{pmatrix} m_1 \\ m_2 \end{pmatrix}$$

Sin embargo, hay una serie de consideraciones acerca de la matriz clave K , que deben tenerse en cuenta:

- Debe ser una matriz cuadrada, de dimensión $n \times n$, donde n es tamaño de n -grama que desea cifrarse.
- Los elementos de la matriz deben ser números enteros del conjunto completo de restos módulo 26. Además, en la medida de lo posible deben asignarse letras a dichos números para formar una palabra clave, más sencilla de recordar que una matriz.
- La matriz K debe tener inversa K^{-1} . Dado que trabajamos dentro de un anillo, la condición anterior debe darse dentro de él, por tanto, el valor del determinante reducido a módulo 26, debe ser distinto de 0.

$$\det(K) \bmod 26 \neq 0$$

- El determinante de la matriz K y el orden del grupo deben ser primos entre sí. Es decir, el máximo común divisor del valor del determinante de la matriz clave y 26 debe ser uno.

$$\text{MCD}(\det(K), 26) = 1$$

A continuación, se cifra el siguiente fragmento de texto, con la matriz clave K , utilizando la siguiente función (A III: Código 12 Cifrado de Hill). Mensaje: “Las cosas podrían haber sucedido de otra manera y, sin embargo, sucedieron así.” De la obra *El camino* de Miguel Delibes.

Clave: FACB, $K = \begin{pmatrix} 5 & 0 \\ 2 & 1 \end{pmatrix}$

A III: Código 1 Asociación 1

Texto Original: LASCOSASPODRIANHABERSUCEDIDODEOTRAMANERAYSINEMBARGOSUCEDIERO NASI

Equivalente numérico: [11, 0, 18, 2, 14, 18, 0, 18, 15, 14, 3, 17, 8, 0, 13, 7, 0, 1, 4, 17, 18, 20, 2, 4, 3, 8, 3, 14, 3, 4, 14, 19, 17, 0, 12, 0, 13, 4, 17, 0, 24, 18, 8, 13, 4, 12, 1, 0, 17, 6, 14, 18, 20, 2, 4, 3, 8, 4, 17, 14, 13, 0, 18, 8]

A III: Código 12 Cifrado de Hill

A III: Código 2 Asociación inversa 1

Cadena de números: [3, 22, 12, 12, 18, 20, 0, 18, 23, 18, 15, 23, 14, 16, 13, 7, 0, 1, 20, 25, 12, 4, 10, 8, 15, 14, 15, 20, 15, 10, 18, 21, 7, 8, 8, 24, 13, 4, 7, 8, 16, 14, 14, 3, 20, 20, 5, 2, 7, 14, 18, 20, 22, 16, 20, 11, 14, 20, 7, 22, 13, 0, 12, 18]

texto asociado: DWMSUASXSPXOQNABUZMEKIPOPUPKSVHIIYNEHIQOODUUFCHOSUWQULOUHW NAMS

Para descifrarlo se debe realizar el mismo proceso, pero con la matriz inversa en módulo 26. En primer lugar, debe comprobarse que en efecto existe la matriz inversa modular. Esto se cumple ya que $\det(K) \bmod 26 = 5 \neq 0$ y por último, $\text{MCD}(5, 26) = 1$. El siguiente código (A III: Código 13 Cálculo matriz inversa) calcula la matriz inversa en módulo 26. Para la matriz K , se obtiene K^{-1} :

A III: Código 13 Cálculo matriz inversa

```
[[21.  0.]
 [10.  1.]]
```

A continuación, para descifrar el mensaje, se utiliza la misma función que para el encriptado (A III: Código 14 Descifrado Hill), pero sustituyendo K por su inversa modular.

A III: Código 14 Descifrado Hill**A III: Código 2 Asociación inversa 1**

Cadena de números: [11, 0, 18, 2, 14, 18, 0, 18, 15, 14, 3, 17, 8, 0, 13, 7, 0, 1, 4, 17, 18, 20, 2, 4, 3, 8, 3, 14, 3, 4, 14, 19, 17, 0, 12, 0, 13, 4, 17, 0, 24, 18, 8, 13, 4, 12, 1, 0, 17, 6, 14, 18, 20, 2, 4, 3, 8, 4, 17, 14, 13, 0, 18, 8]

texto asociado: LASCOSASPODRIANHABERSUCEDIDODEOTRAMANERAYSINEMBARGOSUCEDIERO NASI

Por último, se va a estudiar el criptoanálisis de la cifra de Hill. Para llevarlo a cabo se debe tener en cuenta que donde reside la seguridad de este sistema es en el desconocimiento de la matriz clave K . Un sistema de este tipo no puede ser atacado por fuerza bruta como se hizo con la cifra del César debido al alto número de posibilidades, la única forma de hacerlo es comparando fragmentos de un texto cifrado y el mensaje original equivalente.

Al hacerlo, el criptoanalista debe establecer un sistema de ecuaciones matriciales. Para facilitar esta tarea lo más sencillo es buscar vectores unitarios en alguno de los mensajes.

Los vectores unitarios son aquellos que tienen norma 1. Por ejemplo, en la matriz identidad de dimensión 3, los vectores unitarios serían v_1 , v_2 y v_3 .

$$I = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \left\{ \begin{array}{l} \vec{v}_1 = (100) \\ \vec{v}_2 = (010) \\ \vec{v}_3 = (001) \end{array} \right.$$

Figura 4-7 Vectores unitarios (autoría propia)

Ahora bien, con la asociación de números y letras del alfabeto realizada en este trabajo, los vectores unitarios quedan de la siguiente forma.

$$I = \begin{pmatrix} B & A & A \\ A & B & A \\ A & A & B \end{pmatrix} \left\{ \begin{array}{l} \vec{v}_1 = (BAA) \\ \vec{v}_2 = (ABA) \\ \vec{v}_3 = (AAB) \end{array} \right.$$

Figura 4-8 Vectores unitarios representados por letras (autoría propia)

En un caso práctico, en el supuesto de que se interceptara este criptograma, y se conocieran fragmentos del mensaje original, se deberían localizar vectores unitarios en alguna de las partes para averiguar la matriz de cifrado, y obtener el resto del texto.

Criptograma interceptado:

“LKCSMZQUMACBYXTVPPELJDIEICPCKMBHWKGWMRQUCSIEJLMAEUAYGDHHRHLY
XVEFTBDWQAYSQDGVRSHP CZRNXPZJDRX”

Del cual se conoce el fragmento inicial “TRASLOSCOMBATES...”

Puede obtenerse la matriz de encriptado de la siguiente manera. Se toma como premisa que se conoce que el método utilizado es el criptosistema de Hill, y que la matriz utilizada es de dimensión 2x2. En primer lugar, se comparan ambos mensajes y se buscan vectores unitarios en el texto.

L	K	C	S	M	Z	Q	U	M	A	C	B	Y	X	T
T	R	A	S	L	O	S	C	O	M	B	A	T	E	S

Figura 4-9 Comparación de los textos y búsqueda de vectores unitarios (autoría propia)

En este caso se tiene que (BA) equivale a (CB) en el criptograma, cuyo equivalente numérico según la asociación 1 es C = 2 y B = 1. A continuación resolvemos el sistema.

$$k = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 2 \\ 1 \end{pmatrix}$$

$$a \cdot 1 \bmod 26 = 2;$$

$$a = 2 \cdot \text{inv}(1, 26) \bmod 26 = 2$$

$$c \cdot 1 \bmod 26 = 1;$$

$$c = 1 \cdot \text{inv}(1, 26) = 1$$

De esta manera, se resuelven dos de las incógnitas de la ecuación, a y c . Como puede observarse, la única condición que debe cumplirse a la hora de seleccionar los vectores es que la letra que multiplica la incógnita tenga inverso en el cuerpo de la cifra. Por tanto, no todos los vectores son válidos.

El siguiente vector elegido es (TR), que se corresponde con (LK) en el mensaje original. Este vector es válido para el descifrado ya que T = 19 y R = 17, y se cumple que $MCD(17, 26) = 1$, y por tanto existe inverso de 17 en el cuerpo de la cifra. El sistema sería el siguiente.

$$k = \begin{pmatrix} 2 & b \\ 1 & d \end{pmatrix} \begin{pmatrix} 19 \\ 17 \end{pmatrix} = \begin{pmatrix} 11 \\ 10 \end{pmatrix}$$

$$(2 \cdot 19 + 17 \cdot b) \bmod 26 = 11;$$

$$b = (11 - 38) \cdot \text{inv}(17, 26) \bmod 26 = -27 \cdot -3 \bmod 26 = 3$$

$$(1 \cdot 19 + 17 \cdot d) \bmod 26 = 10;$$

$$d = (10 - 19) \cdot \text{inv}(17, 26) \bmod 26 = -9 \cdot -3 \bmod 26 = 1$$

Por tanto, la matriz clave de este criptosistema sería:

$$K = \begin{pmatrix} 2 & 3 \\ 1 & 1 \end{pmatrix}$$

Y por tanto:

$$K^{-1} = \begin{pmatrix} 25 & 3 \\ 1 & 24 \end{pmatrix}$$

Si desencriptamos el mensaje interceptado, en efecto:

“TRASLOSCOMBATESDELDIADAEYERLAMUNICIONESCASEAYNOMEQUEDAOTRAALTERNATIVAQUEEMPRENDERLARETIRADAX”

4.4 Cifrado por transposición o permutación

Una vez estudiados los sistemas de cifrado por sustitución, se procede al estudio del cifrado por permutación, que es el segundo gran pilar de la criptografía clásica. En este método el mensaje a cifrar se divide en bloques independientes y se reordenan sus caracteres, de tal forma que en texto encriptado se encuentran los mismos símbolos, pero en distinto orden. A este sistema pertenece el uso de la Escítala como se explicó al inicio del trabajo.

De esta manera, un mensaje encriptado en el cual la distribución de caracteres es similar a la característica del lenguaje, es susceptible de haber sido cifrado utilizando la transposición. Por ello, a la hora de llevar a cabo un criptoanálisis, el mensaje no puede atacarse con los mecanismos estudiados hasta ahora, sino que debe recurrirse a una técnica denominada anagramación.

De esta forma, un mensaje $m = m_1, m_2, m_3 \dots$ donde cada m_i es el número entero correspondiente a la asociación establecida en la introducción, se divide en vectores v_j de dimensión n , y se aplica la permutación P .

$$F(v_j, P) = P(v_j) = v_j'$$

Y la función de descifrado.

$$G(v_j', P) = P^{-1}(v_j') = v_j$$

De esta forma, si a un mensaje $m = m_1, m_2, m_3 \dots m_{10}$, se le aplica la siguiente permutación $P_n = 51423$ en periodo 5, el criptograma obtenido es $m' = m_5, m_1, m_4, m_2, m_3, m_{10}, m_6, m_9, m_7, m_8$.

Por ejemplo, si tratamos de cifrar el siguiente mensaje $m = \text{"A BUEN ENTENDEDOR POCAS PALABRAS BASTAN"}$ con la permutación anterior, se obtendría el siguiente mensaje.

$M = \text{ABUEN ENTEN DEDOR POCAS PALAB RASBA STANX}$

$M' = \text{NAEBU NEENT RDOED SPAOC BPAAL ARBAS XSNTA}$

El siguiente fragmento de código (A III: Código 15 Cifrado por permutación) lleva a cabo la permutación de la palabra “algoritmo”, con la permutación $P_n = [2, 1, 3]$.

A III: Código 15 Cifrado por permutación

A III: Código 2 Asociación inversa 1

Cadena de números: [11, 0, 6, 17, 14, 8, 12, 19, 14]

texto asociado: LAGROIMTO

Para llevar a cabo la operación de descifrado se utiliza la siguiente función (A III: Código 16 Cálculo permutación inversa) que se encarga de calcular la permutación inversa, y a continuación bastaría con realizar una nueva permutación para volver al mensaje original.

A III: Código 16 Cálculo permutación inversa

Permutación original: [4, 3, 1, 2]

Permutación inversa: [3, 4, 2, 1]

Los criptogramas obtenidos por permutación pueden ser atacados mediante “fuerza bruta”. Esto, como se vio en el criptosistema del César consiste en intentar descifrar el mensaje utilizando todas las claves posibles. Por tanto, la seguridad de este sistema residirá en la longitud de la clave. Es decir, cuanto mayor sea el periodo de la permutación, mayor es la seguridad, llegando al caso en que la longitud de la clave es igual a la longitud del mensaje.

5 CRIPTOGRAFÍA DE CLAVE PÚBLICA

La criptografía de clave pública o asimétrica utiliza un par de claves para el envío de mensajes. Las dos claves pertenecen a la misma persona, una de ellas es su clave pública, que puede ser entregada y conocida por cualquiera y la otra es su clave privada, de la que solo ella puede tener conocimiento y es la utilizada para descifrar el mensaje. Podría realizarse un símil con entregar candados sin llave, de tal forma que cualquiera puede cerrarlos pero solo el poseedor de la llave es capaz de abrirlos.

5.1 Criptosistema RSA

A continuación se exponen las bases del funcionamiento del encriptado y descifrado del RSA. Para ello se va a utilizar el ejemplo clásico de Alicia, Benito y Eva, Donde Benito desea compartir cierta información con Alicia, pero no quiere que esta sea interceptada por Eva.

Partiendo de la metáfora anterior de los candados, el funcionamiento sería el siguiente. Alicia tiene dos claves, una clave pública, conocida por cualquiera, y una privada que solo conoce ella. La clave pública permite a cualquier usuario, por ejemplo Benito, cifrar un mensaje que solo pueda ser descifrado con la clave privada de Alicia. En otras palabras, Alicia reparte a los usuarios interesados en comunicarse con él candados abiertos, que una vez cerrados solo pueden volver a abrirse con la llave que ella misma posee. Cifrar el mensaje con la clave pública equivale a cerrar el candado, de tal forma que una vez cerrado, ni siquiera el usuario que lo ha hecho sea capaz de abrirlo. En contraposición la clave privada es la llave que abre el candado, que solo tiene Alicia.

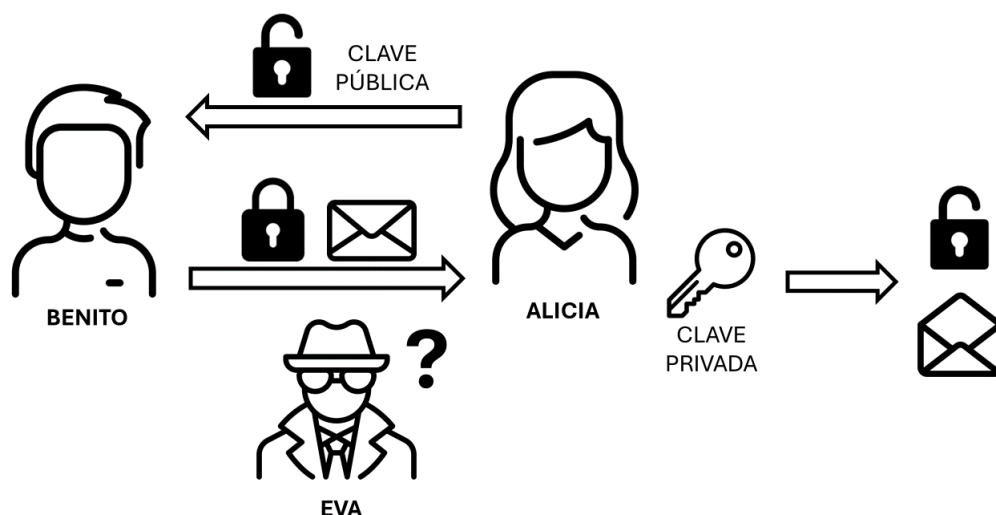


Figura 5-1 Esquema del funcionamiento de los códigos de clave pública (autoría propia)

Utilizando el criptosistema RSA el procedimiento sería el siguiente. Como se ha mencionado, dicho criptosistema emplea dos claves, una clave pública compuesta por los números e y n , y una clave privada formada por d y n . Debido al tiempo de cálculo, sensiblemente mayor que el de un sistema de cifrado simétrico, generalmente se emplea para el intercambio de claves y la firma digital.

En primer lugar se eligen dos números primos p y q . El algoritmo RSA se basa la dificultad de factorizar números primos, por tanto, estos números deben ser de gran tamaño, actualmente del orden de 2048 bits. A continuación se calcula n , resultado del producto entre p y q , es decir, $n = pq$.

Seguidamente se calcula el valor de la función de Euler de n , denotada como $\varphi(n)$. Al ser p y q números primos, se tiene que $\varphi(n) = (p-1) \cdot (q-1)$. Una vez calculado $\varphi(n)$ se escoge un número e cuyas condiciones son que se encuentre en el intervalo $(1, \varphi(n))$ y que exista su inverso multiplicativo. Es necesario que exista inverso multiplicativo porque dicho valor es el número d , que forma parte de la clave privada.

Por último para el cifrado y descifrado se realizan respectivamente las operaciones $C = M^e \pmod{n}$ y $M = C^d \pmod{n}$. Numéricamente el proceso sería de la siguiente manera.

1. En primer lugar, Alicia escoge dos números primos muy grandes, p y q . Para el ejemplo se escogen $p = 83$ y $q = 97$. Estos números solo pueden ser conocidos por Alicia.

2. Alicia calcula el producto de p y q , y obtiene el entero N , donde $N = p \cdot q = 83 \cdot 97 = 8051$. Además elige un número e , tal que e y $\varphi(N) = (p-1) \cdot (q-1)$ deben ser primos entre sí. Es decir, $\text{MCD}(e, \varphi(N)) = 1$, esto es así para que exista el inverso multiplicativo de este número, y por tanto exista d , que será la clave privada. En este caso $\varphi(N) = 7872$ y $e = 11$. El número $\varphi(N)$ es el indicador de Euler, y solo debe ser conocido por el propietario.

3. La clave pública de Alicia está constituida por esos dos números, e y N . Estos pueden ser conocidos por cualquiera.

4. Para encriptar un mensaje M , el contenido del mismo debe ser transformado a números. Se puede utilizar el código ASCII (American Standard Code for Information Interchange). Este código utiliza “palabras” de 7 bits para representar caracteres. El encriptado del mensaje se expresa como $C = M^e \pmod{N}$.

Tabla 1 Equivalencia alfabeto ASCII (autoría propia)

Letra	Número binario	Número decimal	M	1001101	77
			N	1001110	78
A	1000001	65	O	1001111	79
B	1000010	66	P	1010000	80
C	1000011	67	Q	1010001	81
D	1000100	68	R	1010010	82
E	1000101	69	S	1010011	83
F	1000110	70	T	1010100	84
G	1000111	71	U	1010101	85
H	1001000	72	V	1010110	86
I	1001001	73	W	1010111	87
J	1001010	74	X	1011000	88
K	1001011	75	Y	1011001	89
L	1001100	76	Z	1011010	90

5. Supongamos que el mensaje que Benito quiere enviar es “OK”, representado en ASCII por 1001111 1001011, que equivale a 79 y 75. Luego $M = 7975$. Se debe considerar que solo deben cifrarse elementos del cuerpo N , por tanto el tamaño del mensaje M deben ser menor que N , si no lo fuera, este debería ser troceado.

6. A continuación Benito utiliza la clave pública de Alicia, donde, $N = 8051$ y $e = 11$. Por tanto el mensaje encriptado es $C = 7975^{11} \bmod 8051 = 4131$.

7. Benito envía el mensaje $C = 4131$. En este momento, Eva trata de descifrar el mensaje, conoce N y e , y conoce C , sin embargo, es muy complicado sin conocer p y q , desencriptar C y obtener el verdadero valor M .

8. Ahora, Alicia trata de decodificar el mensaje, conociendo p y q . Para ello, calcula d , la clave de la descodificación. El número d debe verificar que $e \cdot d = 1 \bmod [(p-1) \cdot (q-1)]$. En este caso $11 \cdot d = 82 \cdot 96$; $11 \cdot d = 1 \bmod (7872)$. Es decir, d es el inverso modular de e en módulo $(p-1) \cdot (q-1)$. En este caso $d = 2147$.

9. Para desencriptar el mensaje Alicia realiza la siguiente operación. $M = C^d \bmod N$; $M = 4131^{2147} \bmod 8051 = 7975$. Por tanto, Alicia conoce el mensaje “OK”.

Para la generación de claves, es necesario tener en cuenta dos teoremas, que hacen posible el correcto funcionamiento del algoritmo. En primer lugar, se considera el Teorema Chino de los Restos, y en segundo lugar el Pequeño Teorema de Fermat, descritos en los preliminares.

A continuación aparece un fragmento de código que permite realizar el encriptado mencionado anteriormente con el algoritmo RSA. El mensaje a enviar será también $M = \text{“OK”}$, expresión que curiosamente viene de la Guerra de Independencia de los Estados Unidos y significaba “0 Killed”, es decir, que no se había producido ninguna baja en combate.

En primer lugar, se define una función que haga una segunda asociación (A III: Código 17 Asociación 2), en este caso, la del alfabeto latino con el alfabeto ASCII.

A III: Código 17 Asociación 2

Texto Original: OK
Equivalente numérico: 7975

A continuación se necesita una función que genere dos números primos p y q y hagan el cálculo de N , $\phi(N)$ (A III: Código 18 Selección de primos p y q).

A III: Código 18 Selección de primos p y q

p : 911 q : 719 N : 655009 $\phi(N)$: 653380

El resultado depende de la aleatoriedad, es este caso. A continuación se elige el número e que debe ser coprimo de $\phi(N)$, por ejemplo $e = 251$. Se comprueba que $\text{MCD}(e, \phi(N)) = 1$ aplicando el algoritmo de Euclides.

A III: Código 4 Algoritmo de Euclides

$\text{MCD}(251, 653380) : 1$

Se procede al encriptado con la siguiente función (A III: Código 19 Cifrado RSA):

A III: Código 19 Cifrado RSA

Cadena_original: 7975
Cadena_encriptada: 313705

A continuación se calcula d , que es la clave privada, utilizando el algoritmo de Euclides Extendido con el código A III: Código 20 Cálculo de la clave privada d .

```
A III: Código 20 Cálculo de la clave privada d
d: 411291
```

Por último se procede al descifrado con el siguiente programa (A III: Código 21 Descifrado RSA):

```
A III: Código 21 Descifrado RSA
Cadena encriptada: 313705
Cadena_original: 7975
```

Con la siguiente función (A III: Código 22 Asociación 2 inversa) se muestra el mensaje:

```
A III: Código 22 Asociación 2 inversa
Cadena de números: [7975]
texto asociado: OK
```

5.2 Firma digital

Una de las características de los algoritmos de clave pública es que además de permitir el cifrado de mensajes, también permiten garantizar su autenticidad. Para ello, se emplea el mismo algoritmo pero utilizando las claves en orden inverso, es decir, se emplea la clave privada para cifrar un mensaje y la clave pública para descifrarlo. De esta manera, cualquiera es capaz de descifrar el mensaje, pero solo el poseedor de la clave privada es capaz de generarlo, de esta manera se prueba su procedencia.

Por ejemplo, imagínese en un supuesto en el que utilizando las claves anteriores, Alicia quiera establecer una conversación con Benito y deba garantizar que se trata de ella. Para ello, envía un mensaje “A” (A = 65) cifrado con su clave pública.

```
A III: Código 23 Firma Digital
Cadena encriptada: 65
Cadena_original: 496484
```

Y Benito lo descifra, para comprobar que en efecto ha sido Alicia quién le ha enviado el mensaje, dado que solo ella podría haber encriptado el mensaje “A” de esa forma dado que solo ella conoce su clave privada.

```
A III: Código 24 Firma digital comprobación
Cadena_original: 496484
Cadena encriptada: 65
```

En la práctica, aunque es posible cifrar un documento en su totalidad, esta operación se realiza solo sobre un resumen de dicha información, obtenido mediante lo que se denomina un algoritmo de hash. Dicho hash representa de manera única y unívoca la información resumida.

5.3 Ataques al algoritmo RSA

Por último se va a abordar el tema de la seguridad del algoritmo RSA, así como de sus posibles ataques.

5.3.1 Ataque por factorización del módulo

El primer posible ataque es mediante factorización del módulo. Debe tenerse en cuenta que la seguridad de los sistemas de clave pública se basa en funciones trampa. Concretamente, el algoritmo RSA es seguro debido a que la factorización de números primos muy grandes, aunque teóricamente

posible, no lo es en el tiempo de vida útil de un equipo. Esto se debe a que se desconoce como se encuentran distribuidos los números primos.

Por ello, si se encontrara un algoritmo que estableciese la relación entre todos ellos, este tipo de sistemas dejarían de ser seguros, ya que los tiempos computacionales se reducirían drásticamente. De la misma manera, este sistema deja de ser seguro cuando los números utilizados no son de gran tamaño.

5.3.2 Ataque por módulo común

Este ataque es posible cuando se cifra un mismo mensaje M con el mismo módulo N , pero un número e diferente. En este caso, es posible encontrar el mensaje original sin necesidad de factorizar el módulo. Por ejemplo para los mensajes:

$$C = M^e \pmod{N}$$

$$C' = M^{e'} \pmod{N}$$

Si e y e' son coprimos entre sí, lo cual será cierto con bastante probabilidad, entonces existen a y b tales que:

$$a \cdot e + b \cdot e' = 1$$

Entonces:

$$M = C^a \cdot C'^b \pmod{N} = (M^e)^a \cdot (M^{e'})^b \pmod{N} = M^{ae + be'} \pmod{N} = M \pmod{N}$$

5.3.3 Ataque debido a un número e pequeño.

La cantidad de números primos conocidos válidos por sus dimensiones para proporcionar seguridad a un sistema es reducido, por ello cabe la posibilidad de que si varios usuarios comparten el mismo módulo, se tiene que:

$$M^e \pmod{n_1} = C_1$$

$$M^e \pmod{n_2} = C_2$$

$$M^e \pmod{n_k} = C_k$$

Y por el Teorema Chino de los Restos explicado anteriormente, sabemos que este sistema tiene solución M , la cual es congruente en módulo $N = n_1, n_2 \dots n_k$.

5.3.4 Ataque cíclico

El fundamento matemático detrás de este tipo de ataque consiste en que al trabajar dentro de un cuerpo finito de números, si multiplicamos un valor por otro sucesivamente, el algún momento obtendremos el número original.

Por tanto, si ciframos de forma indefinida el mensaje C , cabe recordar que el número e forma parte de la clave pública, de la forma $C' = C^e$ tarde o temprano, volverá a obtenerse el mensaje original, y por tanto, el C_n anterior a dicho mensaje era en efecto el mensaje original M .

5.3.5 Ataque por paradoja de cumpleaños

Las bases de este ataque residen en el concepto poco intuitivo de que si se selecciona una muestra de decenas de personas y se colocan en una fila frente a una habitación, en la cual, en una pizarra se han escrito todos los días del año, y dichas personas entran de uno en uno en el aula y tachan el día de su cumpleaños, la probabilidad de que su fecha ya se encuentre marcada es mayor de el 50% cuando han entrado tan solo 23.

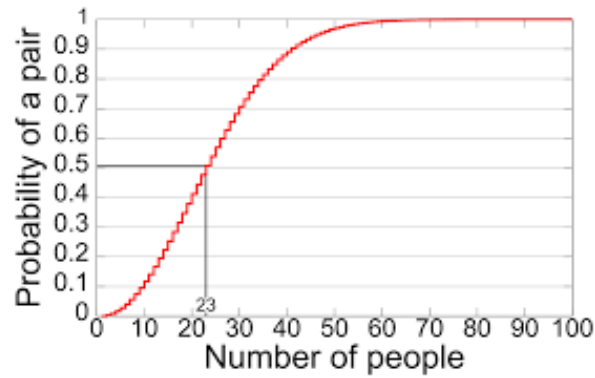


Figura 5-2 Probabilidad de que se produzca una colisión en la paradoja de cumpleaños [9]

Por tanto para una función de encriptado $F(k,m) = m'$, sabemos que es posible encontrar un k' tal que $F(k',m) = m$, y además la probabilidad de alcanzar por azar ese número es superior a la esperada en una primera instancia.

Esta paradoja, aunque en realidad no es tal, sino un resultado poco esperado y de ahí su nombre, funciona en realidad para cualquier sistema de cifrado, no solo para RSA. La seguridad a este ataque consiste en generar las claves lo más extensas posibles.

En conclusión, el sistema RSA sigue siendo seguro y ampliamente empleado hoy en día, sin embargo para que el criptosistema proporcione el margen de confianza adecuado es necesario que este se implemente correctamente. La vulnerabilidad del sistema aumenta cuanto menor es el tamaño de las claves.

6 CRIPTOGRAFÍA CUÁNTICA

Por último, el desarrollo concluye con lo que parece ser el futuro de la criptografía, un sistema que aun se encuentra en desarrollo, pero que en caso de ser implementado, sería teóricamente irrompible. A diferencia del resto de los sistemas que se consideran seguros, estos son teóricamente descifrables, lo que ocurre es que el tiempo de descifrado supera la vida útil del equipo, sin embargo, este sistema sería teóricamente seguro.

Los ordenadores cuánticos utilizan cubits en lugar de bits. En computación clásica un bit puede tomar el valor de 1 o 0, pero en un solo estado al mismo tiempo, mientras que en computación cuántica los cubits pueden encontrarse en infinitos estados al mismo tiempo, este fenómeno se conoce como principio de superposición. Debido a este principio y al de entrelazamiento cuántico, un ordenador cuántico es capaz de realizar el mismo número de cálculos que un ordenador actual en un tiempo sensiblemente menor. Actualmente, los ordenadores cuánticos se encuentran aun en desarrollo. Empresas como Google o IBM se encuentran trabajando en el desarrollo de los mismos.

La existencia de un ordenador cuántico desarrollado tendría un impacto significativo en el ámbito de la criptología. Su velocidad de cálculo haría que criptosistemas que hasta ahora se consideran seguros y en los que se basa la seguridad de la información actual pudieran atacarse por fuerza bruta. En referencia a esto debe mencionarse el algoritmo de Shor, capaz teóricamente de resolver el problema de la factorización de números enteros en tiempo polinómico en un ordenador cuántico, mientras que en uno clásico podría hablarse de miles de años.

Los protocolos desarrollados en el ámbito de la criptografía clásica están orientados a la distribución de claves. El concepto de distribución de claves cuánticas QKD (Quantum Key Distribution) fue propuesto en la década de los 70 por Stephen Wiesner, sin embargo el primer protocolo fue propuesto en 1984 conocido como BB84. [25] A continuación se procede a explicar el funcionamiento de la distribución de claves cuánticas, de nuevo a través de Alicia, Benito y Eva.

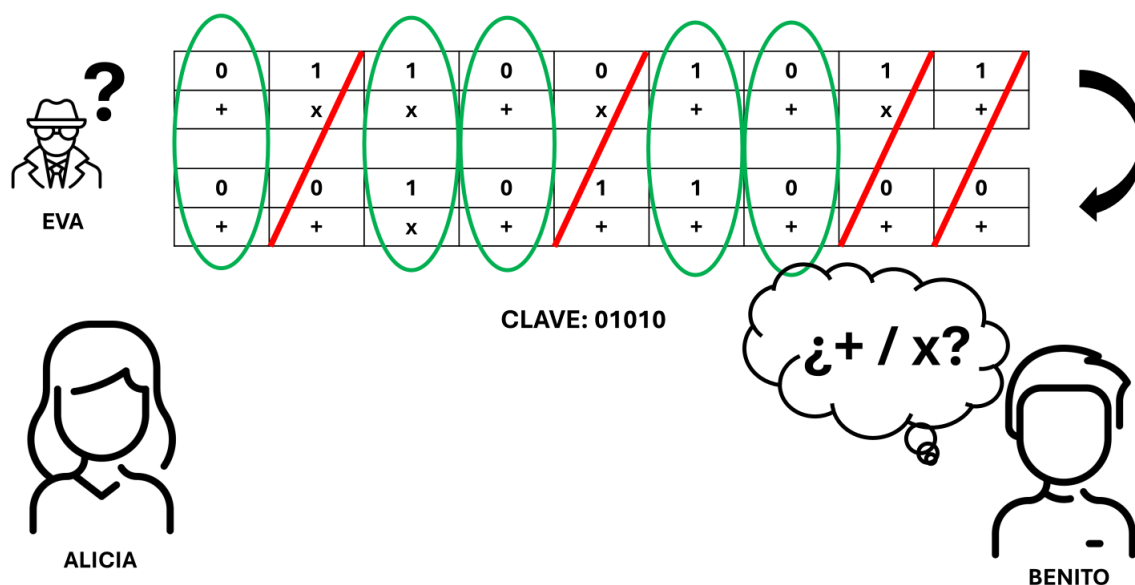


Figura 6-1 Esquema del protocolo QKD (autoría propia)

En primer lugar Alicia genera una secuencia aleatoria de bits, es decir, unos y ceros, y los transforma en cubits. Estos cubits se envían a través de un medio capaz de transmitir partículas cuánticas, como por ejemplo fotones, a través de un cable de fibra óptica. Estos fotones se encuentran polarizados también de forma aleatoria. Esta polarización puede ser rectilínea (+) o diagonal (X).

A continuación estos fotones se envían a Benito quien los recibe, aunque no sabe con que base se polarizaron. Benito realiza la medición utilizando las bases de forma aleatoria, con una probabilidad de acertar de $\frac{1}{2}$. Benito ahora tiene una serie de bits, pero no sabe cuales fueron polarizados con la base que el utilizó como filtro y cuales no.

Por ello, Alicia y Benito contactan a través de un canal público y se transmiten las bases que utilizaron para polarizar los fotones. Por último toman los bits en los cuales ambos utilizaron la misma base y descartan el resto. Ese número de bits que solo ellos conocen es su clave secreta.

En caso de que Eva intentara obtener información del estado de los fotones, al observarlos perturbaría el canal y haría que estos se comportaran de manera distinta. Esto hace que la tasa de errores en las bases utilizadas aumente, lo que hará que Alicia y Benito sospechen que Eva está intentando interferir sus comunicaciones y aborten la transmisión insegura en curso. [26]

7 CONCLUSIONES Y LÍNEAS FUTURAS

7.1 Conclusiones

Los objetivos de este trabajo eran, en primer lugar, trazar la evolución de los códigos a través de la Historia. Además, se pretendía exponer la importancia de la criptografía hoy en día.

A lo largo del estado del arte, se estableció una línea temporal en la cual se mencionaban los hitos más importantes en la historia de la criptología, destacando aquellos libros o artículos en los cuales se encontraban publicados los criptosistemas de mayor relevancia. Además, durante el desarrollo, el estudio detallado de cada uno de los códigos ha facilitado la comprensión de como dicha evolución sucedió de forma progresiva, a lo largo de un proceso adaptativo en el cual cada criptosistema era criptoanalizado, a veces durante siglos, resuelto y posteriormente reemplazado por uno más robusto que el anterior. También se ha destacado como la evolución de la computación está íntimamente ligada a la criptografía, de tal forma que el desarrollo de los mismos ha permitido atacar y crear criptosistemas cada vez más complejos, hasta llegar a la situación actual en la cual las grandes empresas informáticas se encuentran trabajando en la computación cuántica.

Además, se ha analizado como en el mundo actual, en la Era de la Información, la criptografía tiene un papel esencial en la defensa de dicha información, garantizando la privacidad y la libertad en la red de los usuarios, protegiendo sus datos, almacenados en bases de datos de todo tipo, desde empresas de venta telemática como Amazon, hasta compañías proveedoras de servicios de salud, pasando por bancos. Demostrando que la criptografía no es una ciencia que pertenezca al pasado, sino que es ahora más importante que nunca.

Por lo expuesto anteriormente, se considera que los objetivos de este trabajo de investigación han sido alcanzados, y que tanto el alumno como cualquier lector con acceso al mismo tras la lectura del documento tienen una idea generalizada pero consistente de como funciona los algoritmos actuales de encriptado y como se ha alcanzado la situación actual, así como de la relevancia e impacto de los mismos.

7.2 Líneas futuras

Tras realizar un estudio de los criptosistemas más relevantes a lo largo de la Historia, se ha concluido que las líneas futuras de investigación podrían desarrollarse en las siguientes direcciones.

7.2.1 Criptografía aplicada a defensa

Como se ha visto a lo largo del trabajo, la evolución en el ámbito criptográfico ha venido motivada en muchas ocasiones por la necesidad de compartir información de carácter militar. Los primeros mensajes ocultos enviados por los griegos utilizaban técnicas de esteganografía, Julio César cifraba sus mensajes con códigos elaborados por él mismo, la Cifra de Alberti nació de la necesidad de El Vaticano

de encriptar sus comunicados, en una época en que la Santa Sede disponía de fuerza militar, la máquina Enigma fue desarrollada por los alemanes tras la Primera Guerra Mundial y Alan Turing tuvo que enfrentarse a ella durante el desarrollo de la Segunda Guerra Mundial para proporcionar al bando aliado una posición ventajosa.

Hoy en día el ámbito de defensa sigue requiriendo de sistemas de encriptado robustos, tanto a nivel estratégico de carácter computacional para guardar su información, como a un nivel táctico, en el campo de batalla para proporcionar mando y control a las unidades de combate y a sus comandantes. Por ejemplo la empresa Thales Group se encarga del cifrado de muchos de los equipos radio de HF (High Frequency), VHF (Very High Frequency) y UHF (Ultra High Frequency) utilizados hoy en día por países de la OTAN y la Unión Europea. Por ello una posible línea de investigación futura sería el estudio de los criptosistemas aplicados al ámbito específico del Ministerio de Defensa.



Figura 7-1 JTAC del US Air Force realizando el guiado terminal de una aeronave a través de enlace VHF [27]

7.2.2 Criptografía cuántica

Las líneas de desarrollo actuales en el ámbito criptográfico tienden hacia la computación cuántica. La velocidad de desarrollo de la tecnología actualmente es vertiginosa, de tal forma que los avances tecnológicos se producen en un ritmo exponencial. La concepción y tamaño actuales de un ordenador eran impensables hace cincuenta años. La ley de Moore estima que cada dos años se duplica el número de transistores de un microprocesador [28]. Por este motivo un trabajo subsiguiente podría ir encaminado hacia las técnicas de encriptación que se encuentran en desarrollo actualmente y que en un futuro podrían emplearse de manera regular.

8 BIBLIOGRAFÍA

- [1] Real Academia Española, «Definición de criptografía,» de *Diccionario de la lengua española*.
- [2] Y. L. Jonathan Katz, *Introduction to Modern Cryptography*, Florida, EEUU: Chapman & Hall, 2012.
- [3] M. C. Oliván, *La era de la información (vol,1): economía, sociedad y cultura. La sociedad red*, Barcelona, España: Alianza Editorial, 2005.
- [4] J. G. Carmona, *Tratado de criptografía con aplicación especial al Ejército*, Madrid, España: Sucesores de Rivadeneyra, 1894.
- [5] A. Kerchoffs, *La cryptographie militaire*, *Journal des sciences militaires*, 1883.
- [6] J. Pons, «Sistemas para ciberdefensa,» *Revista Española de Defensa n° 383*, 2021.
- [7] A. C. N. C. G. C. T. D. L. F. F. M. M. M. M. C. P. N. M. T. E. B. J. C. M. F. A. G. H. T. Paul Zimmermann, *Computational Mathematics with SageMath*, Creative Commons, 2018.
- [8] S. Singh, *Los códigos secretos*, Madrid: DEBATE, 2000.
- [9] J. R. A. Alfonso Muñoz Muñoz, *Cifrado de las comunicaciones digitales. De la cifra clásica al algoritmo RSA.*, Madrid: oxWord, 2013.
- [10] I. A. Al-Kadit, *Origins of cryptology: the arab contributions*, *Cryptologia*, vol. 16, Riyadh, ASUDI ARABIA, 1992.
- [11] M. J. Prieto, *Historia de la Criptografía*, Madrid: la esfera de los libros, 2020.
- [12] A. R. González, «Conferencia sobre Jaime Janer Robinson,» Marín, Pontevedra, 2024.
- [13] Universidad de Valladolid, «Tema 1. ¿Qué es un ordenador?,» de *Evolución histórica de los ordenadores*.
- [14] Á. Hernández, «¿Cuánto tardaría un ordenador actual en romper el código Enigma de los nazis?,» *El confidencial*, 29 06 2015.
- [15] W. A. Kotas, «A Brief Jistory of Cryptography,» University of Tennessee, Knoxville, 2000.

- [16] Marine Corps University, «usmcu,» 2006. [En línea]. Available: <https://www.usmcu.edu/Research/Marine-Corps-History-Division/People/Navajo-Code-Talkers-in-WWII/>. [Último acceso: 17 03 2024].
- [17] L. S. Hill, «Cryptography in an Algebraic Alphabet,» *The American Mathematical Monthly*, vol. 36, pp. 306-312, 1929.
- [18] C. E. Shannon, «A Mathematical Theory of communication,» *The Bell System Technical Journal*, vol. 27, pp. 379-423, 623-656, 1948.
- [19] C. E. Shannon, «Communication Theory of Secrecy Systems,» *Bell System Technical Journal*, vol. 28, pp. 656-715, 1949.
- [20] M. E. H. Whitfield Diffie, «New Directions in Cryptography,» *Transactions on Information Theory*, vol. 22, nº 6, 1976.
- [21] S. Cook, «The P versus NP Problem,» *Clay Mathematics Institute, University of Toronto*, 2000.
- [22] Á. d. R. Mateos, «Introducción a la Criptología,» Murcia, 2021.
- [23] D. Traill, «DocsTeach, The Night Before D-Day,» National Archives, National Archives Foundation, [En línea]. Available: <https://www.docsteach.org/activities/teacher/the-night-before-dday>. [Último acceso: 04 04 2024].
- [24] America's National Churchill Museum, [En línea]. Available: <https://www.nationalchurchillmuseum.org/never-give-in-never-never-never.html>. [Último acceso: 02 04 2024].
- [25] S. Wiesner, «Conjugate Coding,» *Columbia University, New York*, 1970.
- [26] M. N. Antonio Acín, «Criptografía cuántica,» *Revista Española de Física*, 2008.
- [27] M. Tan, «Army, Air Force formalize program putting JTACs in Army units,» *ArmtTimes*, 2015.
- [28] G. E. Moore, «Cramming more components onto integrated circuits,» Intel, 2005.
- [29] S. Morgan, «Top 10 Predictions And Statistics for 2024,» *Cybercrime Magazine*, 5 febrero 2024. [En línea]. Available: <https://cybersecurityventures.com/top-5-cybersecurity-facts-figures-predictions-and-statistics-for-2021-to-2025/>. [Último acceso: 19 marzo 2024].
- [30] A. S. Cerdeña, «Ciberética y sociedad de la información,» *Revista de Filosofía*, nº 33, 2004.

ANEXO I: IMPLICACIONES SOCIALES, ECONÓMICAS Y AMBIENTALES

En el espectro social, económico y ambiental, la criptografía también repercusiones, y analizar estas es el objeto de este anexo.

A nivel social, aunque en el siguiente anexo se expone con mayor detalle, cabe destacar que la criptografía garantiza a los ciudadanos la seguridad de su información. Información relevante como el historial médico o el número de cuenta bancario se encuentra almacenada en diferentes servidores en la red y es gracias a la existencia de sistemas de cifrado que la sociedad puede almacenar su información y comunicarse utilizando tecnología de la información con la garantía de que esta no es fácilmente accesible y usada de forma indebida.

En cuanto al ámbito económico, la criptografía toma un papel fundamental a la hora de realizar transacciones y pagos de forma telemática, preservar la información de las cuentas bancarias de los clientes y proteger los activos digitales de las empresas. Por ejemplo, a la hora de realizar transacciones financieras u otras operaciones comerciales es fundamental verificar la identidad de ambas partes, para ello se utilizan protocolos de autenticación basados en la firma digital, que permite por ejemplo el algoritmo RSA. En este aspecto, las criptomonedas, desde su aparición en 2009 han ofrecido una alternativa en este aspecto, permitiendo realizar transacciones rápidas, sin recurrir a terceras partes y de forma descentralizada, ofreciendo un mercado en crecimiento pero también de carácter volátil. Por último, no debe pasar desapercibido el impacto económico que tiene la ciberdelincuencia en la economía. Se estima que el valor de los ciberdelitos crecerá un 15% anualmente en los próximos dos años, alcanzando la cifra de 9.5 trillones de dólares americanos en 2024 y 10.5 trillones de dólares americanos en 2025, respecto de los 3 trillones de dólares en 2015 [29]. De ser así su valor correspondería al de la tercera economía más importante del mundo tras EEUU y China. Además, también se estima que a finales de este año existan 3.5 millones de puestos de trabajo relativos a la ciberseguridad. [29]

Finalmente en cuanto a las consecuencias medioambientales, es difícil encontrar una relación directa entre la ciberseguridad y su impacto en la sostenibilidad del medio ambiente. Sin embargo, la criptografía si que es necesaria y se utiliza para guardar y tratar los datos referentes al medio ambiente para garantizar su integridad. Además, se usa para proteger los servidores de datos pertenecientes a instalaciones tales como plantas nucleares o empresas farmacéuticas lo que ayuda a prevenir potenciales ataques de carácter NBQ (Nuclear Biológico Químico).

ANEXO II: REFLEXIONES ÉTICAS Y SOCIALES

En relación con las implicaciones éticas y sociales de la criptografía, cabe mencionar que en esta época la información es más valiosa que nunca, y su intercambio digital un pilar de nuestra sociedad. Cada día se envían millones de correos electrónicos y gestiones de la administración pública o movimientos bancarios son realizados de forma telemática. En todos estos procesos es necesario el uso de la criptografía. Afortunadamente, los sistemas criptográficos se encuentran ahora más disponibles que nunca. Sin embargo, si la codificación es capaz de proteger la información de la ciudadanía y prestar un servicio a la sociedad defendiendo la privacidad y derechos de sus individuos, también puede usarse para llevar a cabo acciones ilícitas o actos contra el gobierno. Esta situación da pie al debate entre la libertad del individuo y la seguridad del colectivo, un debate que sigue abierto hoy en día.

Por otro lado, previamente se ha expuesto el impacto de las tecnologías de información y comunicaciones en nuestra sociedad. Este es tal, que al hablar de esta tecnología es inevitable mencionar la criptografía y seguridad de las comunicaciones y unido a estas la ciberética.

El nacimiento de la ciberética se sitúa en 1950 con las reflexiones de carácter moral realizadas por Norbert Wiener en su obra *The Human Use of Human Beings*. Desde entonces las reflexiones en este ámbito han seguido desarrollándose dando lugar a la publicación de artículos y tratados referentes a este tema como por ejemplo la declaración del consejo de Derechos Humanos de la ONU sobre derechos humanos e internet. La ONU menciona principalmente la libertad de expresión y el derecho a la privacidad, así como el derecho a la no discriminación. [30]

La criptografía debe por tanto entenderse como una herramienta al servicio de la sociedad. Una forma de garantizar que se respeten los derechos humanos en el ciberespacio garantizando la privacidad de la información de los usuarios y su libertad de expresión. Como se ha destacado a lo largo del trabajo la criptografía está presente en nuestro quehacer diario, sin ella no podríamos utilizar servicios de mensajería en línea, realizar llamadas por teléfono de forma segura, consultar el estado de nuestra cuenta corriente o realizar pagos por internet. Es decir, la sociedad actual no puede entenderse sin la criptografía y por ello, sobre los criptógrafos recae la responsabilidad de indagar en el desarrollo de la misma de tal forma que la gestión de la información y el intercambio de esta sigan siendo seguros.

ANEXO III: CÓDIGOS

A III: Código 1 Asociación 1

```
import string

#La siguiente función crea un diccionario para asociar cada letra del alfabeto a un número de 0 a 25.
def asociacion_letras_numeros():
    alfabeto_mayusc = string.ascii_uppercase
    asociacion1 = {}
    for indice, letra in enumerate(alfabeto_mayusc):
        asociacion1[letra] = indice
    return asociacion1

#La siguiente función itera a través de cada caracter del texto y añade su número a la lista 'texto_numeros'.
def convertir_texto_numeros(texto, asociacion1):
    texto_numeros = []
    for letra in texto:
        if letra.isalpha() and letra.isupper():
            numero = asociacion1[letra]
            texto_numeros.append(numero)
        else:
            raise ValueError("El texto debe ser en mayusculas y sin espacios")
    return texto_numeros

try:
    texto_original = "ESTAISAPUNTODEEMBARCAROSENLAGRANCRUZADALOSOJOSDELMUNDOOSCONTEMPLAN"
    asociacion_abecedario = asociacion_letras_numeros()
    texto_numeros = convertir_texto_numeros(texto_original, asociacion_abecedario)
    print("Texto Original:", texto_original)
    print("Equivalente numérico:", texto_numeros)
except ValueError as e:
    print(e)
```

A III: Código 2 Asociación inversa 1

```
import string

#Esta función crea el mismo diccionario que en el primer caso, pero en dirección inversa.
def asociacion_numeros_letras():
    alfabeto_mayusc = string.ascii_uppercase
    asociacion1 = {}
    for indice, letra in enumerate(alfabeto_mayusc):
        asociacion1[indice] = letra
    return asociacion1

#Esta función asocia la letra correspondiente a cada número de la lista.
def convertir_numeros_texto(numeros, asociacion1):
    lista_letras = []
    for numero in numeros:
        letra = asociacion1[numero]
        lista_letras.append(letra)
    return ''.join(lista_letras)

cadena_de_numeros = [11, 20, 2, 7, 0, 17, 4, 12, 14, 18, 4, 13, 11, 0, 18, 15, 11, 0, 24, 0, 18, 11, 20, 2, 7, 0, 17, 4, 12, 14, 18, 4, 13, 11, 0, 18, 15, 8, 18, 19, 0, 18, 3, 4, 0, 19, 4, 17, 17, 8, 25, 0, 9, 4, 11, 20, 2, 7, 0, 17, 4, 12, 14, 18, 4, 13, 11, 14, 18, 2, 0, 12, 15, 14, 18, 24, 4, 13, 11, 0, 18, 2, 0, 11, 11, 4, 18, 11, 20, 2, 7, 0, 17, 4, 12, 14, 18, 4, 13, 11, 0, 18, 2, 14, 11, 8, 13, 0, 18, 13, 20, 13, 2, 0, 13, 14, 18, 17, 4, 13, 3, 8, 17, 4, 12, 14, 18]

asociacion_abecedario_inversa = asociacion_numeros_letras()
texto_asociado = convertir_numeros_texto(cadena_de_numeros, asociacion_abecedario_inversa)
print("Cadena de números:", cadena_de_numeros)
print("texto asociado:", texto_asociado)
```

A III: Código 3 Contador de letras

```
#Esta función crea un diccionario en el que almacena la frecuencia de cada
letra.
def contar_letras(mensaje):
    frecuencia_letras = {}
    for caracter in mensaje:
        if caracter.isalpha():
            letra = caracter.upper()
            frecuencia_letras[letra] = frecuencia_letras.get(letra, 0) + 1
    return frecuencia_letras

#Esta función imprime el alfabeto y la frecuencia de cada letra.
def lista_frecuencias(frecuencia):
    alfabeto = 'ABCDEFGHIJKLMNOPQRSTUVWXYZ'
    for letra in alfabeto:
        cantidad = frecuencia.get(letra, 0)
        print(f"{letra}:{cantidad}")

mensaje = "Apurar, cielos, pretendo ya que me tratáis así, qué delito cometí
contra vosotros naciendo; aunque si nací, ya entiendo qué delito he cometido
. Bastante causa ha tenido vuestra justicia y rigor; pues el delito mayor de
l hombre es haber nacido."
resultado = contar_letras(mensaje)
lista_frecuencias(resultado)
```

A III: Código 4 Algoritmo de Euclides

```
#El siguiente código realiza las iteraciones del algoritmo de Euclides hasta
que a = b, en ese caso devuelve el valor a que es el MCD.
def algoritmo_Euclides(a, b):
    while a != b:
        if a > b:
            a = a - b
        if a < b:
            b = b - a
    return a

A = 195
B = 52
MCD = algoritmo_Euclides(A, B)
print (f"MCD({A},{B}):{MCD}")
```

A III: Código 5 Algoritmo de Euclides Extendido

```
#El siguiente código realiza las iteraciones del algoritmo de Euclides extendido hasta que b = 0, en ese caso devuelve el valor a que es el MCD, el de u y el de v.
def algoritmo_Euclides_extendido(a, b):

    u0 = 1
    v0 = 0
    u1 = 0
    v1 = 1

    while b != 0:
        q = a//b
        r = a - b * q
        u = u0 - q * u1
        v = v0 - q * v1
        a = b
        b = r
        u0 = u1
        v0 = v1
        u1 = u
        v1 = v
    return a, u0, v0

A = 78
B = 32
resultado = algoritmo_Euclides_extendido(A, B)
print ("MCD:", resultado[0], "u:", resultado[1], "v:", resultado[2])
```

A III: Código 6 Cifra del César

```
#Esta función toma los valores de la lista de números, suma k y almacena el nuevo valor.
def cifrado_cesar(cadena_de_numeros, desplazamiento):
    cadena = []
    for numero in cadena_de_numeros:
        numero = (numero + desplazamiento) % 26
        cadena.append(numero)
    return cadena

cadena_de_numeros = [0, 11, 4, 0, 8, 0, 2, 19, 0, 4, 18, 19]
desplazamiento = 3

cadena_encriptada = cifrado_cesar(cadena_de_numeros, desplazamiento)
print("Cadena inicial:", cadena_de_numeros)
print("Cadena encriptada:", cadena_encriptada)
```

A III: Código 7 Descifrado César

```
#Esta función toma los valores de la lista de números, resta k y almacena el
nuevo valor.
def descifrado_cesar(cadena_de_numeros, desplazamiento):
    nueva_cadena = []
    for numero in cadena_de_numeros:
        numero = numero - desplazamiento % 26
        nueva_cadena.append(numero)
    return nueva_cadena

cadena_de_numeros = [3, 14, 7, 3, 11, 3, 5, 22, 3, 7, 21, 22]
desplazamiento = 3

cadena_descifrada = descifrado_cesar(cadena_de_numeros, desplazamiento)
print("Cadena encriptada:", cadena_de_numeros)
print("Cadena original:", cadena_descifrada)
```

A III: Código 8 Criptoanálisis César por fuerza bruta

```
#Este código toma una lista de números y prueba todos los posibles desplazam
ientos para un valor k entre 0 y 25.
def descifrado_cesar_fuerza_bruta(cadena_de_numeros):
    cadenas = []
    for desplazamiento in range(26):
        cadena = []
        for numero in cadena_de_numeros:
            numero = (numero + desplazamiento) % 26
            cadena.append(numero)
        cadenas.append(cadena)
    return cadenas

cadena_de_numeros = [3, 14, 7, 3, 11, 3, 5, 22, 3, 7, 21, 22]
posibles_mensajes = descifrado_cesar_fuerza_bruta(cadena_de_numeros)
print(posibles_mensajes)
```

A III: Código 9 Cifra de Vigenère

```
#Esta función une dos cadenas (mensaje y clave) y las suma.
#clave2 es la palabra clave repetida con la longitud adecuada.
def cifrado_vigenere(cadena_numeros, clave):
    reps = len(cadena_numeros) / len(clave)
    reps_redondeo_alto = math.ceil(reps)
    clave2 = clave*(reps_redondeo_alto)
    union_cadenas = zip(cadena_numeros, clave2)
    nueva_cadena = [sum(elementos) % 26 for elementos in union_cadenas]
    return nueva_cadena

mensaje_original = [19, 14, 3, 14, 0, 17, 19, 4, 3, 4, 11, 0, 6, 20, 4, 17,
17, 0, 18, 4, 1, 0, 18, 0, 4, 13, 4, 11, 4, 13, 6, 0, 13, 14]
k = [18, 20, 13, 19, 25, 20]
texto_cifrado = cifrado_vigenere(mensaje_original, k)

print ("mensaje original:", mensaje_original)
print ("mensaje cifrado:", texto_cifrado)
```

A III: Código 10 Descifrado Vigenère

```
#Esta función hace negativo el valor de cada número de la lista clave.
def negar_clave(clave):
    clave_negativa = [-a for a in clave]
    return clave_negativa

#IDEM a la función de cifrado, pero al ser la clave negativa el valor resta.
def descifrado_vigenere(cadena_numeros, clave):
    reps = len(cadena_numeros) / len(clave)
    reps_redondeo_alto = math.ceil(reps)
    clave2 = clave*(reps_redondeo_alto)
    union_cadenas = zip(cadena_numeros, clave2)
    nueva_cadena = [sum(elementos) % 26 for elementos in union_cadenas]
    return nueva_cadena

mensaje_original = [11, 8, 16, 7, 25, 11, 11, 24, 16, 23, 10, 20, 24, 14, 17
, 10, 16, 20, 10, 24, 14, 19, 17, 20, 22, 7, 17, 4, 3, 7, 24, 20, 0, 7]
k = [18, 20, 13, 19, 25, 20]
clave_negada = negar_clave(k)
texto_cifrado = descifrado_vigenere(mensaje_original, clave_negada)

print ("mensaje cifrado:", mensaje_original)
print ("mensaje descifrado:", texto_cifrado)
```

A III: Código 11 Método Kasiski

```
#Esta función divide el texto en subtextos en función del tamaño de la clave
. Con este código el tamaño máximo de la clave sería cinco, pero pueden añadirse más listas.
def formar_submensajes(mensaje, tamaño_clave):
    Ma = []
    Mb = []
    Mc = []
    Md = []
    Me = []
    for indice, letra in enumerate(mensaje):
        if (indice + 1) % tamaño_clave == 1:
            Ma.append(letra)
        if (indice + 1) % tamaño_clave == 2:
            Mb.append(letra)
        if (indice + 1) % tamaño_clave == 3:
            Mc.append(letra)
        if (indice + 1) % tamaño_clave == 4:
            Md.append(letra)
        if (indice + 1) % tamaño_clave == 0:
            Me.append(letra)

    print(f"Ma: {''.join(Ma)}, Mb: {''.join(Mb)}, Mc: {''.join(Mc)},Md: {''.join(Md)}, Me: {''.join(Me)} ")

mensaje = "YFAGNLORBJEDDYNBYLHMISHJPEGJHBYJHDBYIRSDYRNZOEFXIERUTBMIDRJFGBM
INDDYNBUFEIJBURBISFXFOYXMEFYJBBXENIUBNLBEQNMUFXDFOBINOQDLOFLOAYMISRBWAEJVAVX
MDRLLIFCULANARB"
tamaño_clave = 4
Mensajes = formar_submensajes(mensaje, tamaño_clave)
```

A III: Código 12 Cifrado de Hill

```
#importa la libreria Numnpy.
import numpy

#Agrupar los elementos de la lista en grupos de dos o tres componentes, según
n.
def agrupar_en_vectores(mensaje, n):
    mensaje2 = []
    if n == 3:
        for i in range(0, len(mensaje), n):
            nuevo_i = [mensaje[i], mensaje[i + 1], mensaje[i + 2]]
            mensaje2.append(nuevo_i)
        return mensaje2
    if n == 2:
        for i in range(0, len(mensaje), n):
            nuevo_i = [mensaje[i], mensaje[i + 1]]
            mensaje2.append(nuevo_i)
        return mensaje2

#Con los grupos anteriores forma vectores.
def convertir_vectores(mensaje2):
    vectores = []
    for sublista in mensaje2:
        v = Matrix(sublista)
        vectores.append(v)
    return vectores

#Esta función multiplica la matriz K por cada uno de los vectores, obteniend
o los vectores ya cifrados.
def cifrado_de_hill(vectores, K):
    vectores_cifrados = []
    for vector in vectores:
        vc = K * vector.transpose()
        vectores_cifrados.append(vc)
    return vectores_cifrados

#Toma los vectores y los une en una lista de números, posteriormente aplica
módulo 26.
def lista_de_numeros(vectores_cifrados):
    lista_final = []
    for vector in vectores_cifrados:
        elemento = vector
        lista_final.extend(elemento)
        lista_final2 = [elem[0] for elem in lista_final]
        lista_final3 = [numero % 26 for numero in lista_final2]
    return lista_final3

n = 2
m = [11, 0, 18, 2, 14, 18, 0, 18, 15, 14, 3, 17, 8, 0, 13, 7, 0, 1, 4, 17,
18, 20, 2, 4, 3, 8, 3, 14, 3, 4, 14, 19, 17, 0, 12, 0, 13, 4, 17, 0, 24, 18,
8, 13, 4, 12, 1, 0, 17, 6, 14, 18, 20, 2, 4, 3, 8, 4, 17, 14, 13, 0, 18, 8]
K = Matrix([[5, 0], [2, 1]])
mensaje2 = agrupar_en_vectores(m, n)
vectores = convertir_vectores(mensaje2)
vectores_cifrados = cifrado_de_hill(vectores, K)
lista_final = lista_de_numeros(vectores_cifrados)
print (lista_final)
```

A III: Código 13 Cálculo matriz inversa

```
#Esta función calcula la matriz inversa en módulo 26, utilizando el algoritmo de Euclides extendido.
def matriz_inversa_modular(K):
    kdet = numpy.linalg.det(K)
    kinv = numpy.linalg.inv(K)
    kadj = kdet * kinv

    a = math.ceil(kdet)
    b = 26
    u0 = 1
    v0 = 0
    u1 = 0
    v1 = 1

    while b != 0:
        q = a//b
        r = a - b * q
        u = u0 - q * u1
        v = v0 - q * v1
        a = b
        b = r
        u0 = u1
        v0 = v1
        u1 = u
        v1 = v
    inv_det_26 = u0 % 26
    kinv_mod = (inv_det_26 * kadj) % 26

    return kinv_mod

K = Matrix([[5, 0], [2, 1]])
solucion = matriz_inversa_modular(K)
print (solucion)
```

A III: Código 14 Descifrado Hill

```
#importa la libreria Numnpy.
import numpy

#Agrupar los elementos de la lista en grupos de dos o tres componentes, según
n.
def agrupar_en_vectores(mensaje, n):
    mensaje2 = []
    if n == 3:
        for i in range(0, len(mensaje), n):
            nuevo_i = [mensaje[i], mensaje[i + 1], mensaje[i + 2]]
            mensaje2.append(nuevo_i)
        return mensaje2
    if n == 2:
        for i in range(0, len(mensaje), n):
            nuevo_i = [mensaje[i], mensaje[i + 1]]
            mensaje2.append(nuevo_i)
        return mensaje2

#Con los grupos anteriores forma vectores.
def convertir_vectores(mensaje2):
    vectores = []
    for sublista in mensaje2:
        v = Matrix(sublista)
        vectores.append(v)
    return vectores

#Esta función multiplica la matriz K por cada uno de los vectores, obteniend
o los vectores ya descifrados.
def descifrado_de_hill(vectores, K):
    vectores_cifrados = []
    for vector in vectores:
        vc = K * vector.transpose()
        vectores_cifrados.append(vc)
    return vectores_cifrados

#Toma los vectores y los une en una lista de números, posteriormente aplica
módulo 26.
def lista_de_numeros(vectores_cifrados):
    lista_final = []
    for vector in vectores_cifrados:
        elemento = vector
        lista_final.extend(elemento)
        lista_final2 = [elem[0] for elem in lista_final]
        lista_final3 = [numero % 26 for numero in lista_final2]
    return lista_final3

n = 2
m = [3, 22, 12, 12, 18, 20, 0, 18, 23, 18, 15, 23, 14, 16, 13, 7, 0, 1, 20,
25, 12, 4, 10, 8, 15, 14, 15, 20, 15, 10, 18, 21, 7, 8, 8, 24, 13, 4, 7, 8,
16, 14, 14, 3, 20, 20, 5, 2, 7, 14, 18, 20, 22, 16, 20, 11, 14, 20, 7, 22, 1
3, 0, 12, 18]
K = Matrix([[21, 0], [10, 1]])
mensaje2 = agrupar_en_vectores(m, n)
vectores = convertir_vectores(mensaje2)
vectores_cifrados = descifrado_de_hill(vectores, K)
lista_final = lista_de_numeros(vectores_cifrados)
print (lista_final)
```

A III: Código 15 Cifrado por permutación

```
#Esta función define la longitud de la clave.
def longitud_clave(clave):
    n = len (clave)
    return n

#Agrupar los elementos de la lista en grupos de n componentes, según el tamaño de la clave.
def agrupar_en_vectores(mensaje, n):
    mensaje2 = []
    for i in range (0, len(mensaje), n):
        nuevo_i = mensaje[i:i+n]
        mensaje2.append(nuevo_i)
    return mensaje2

#Esta función es la que lleva a cabo la permutación de los valores de cada vector, atendiendo a los valores de Clave.
def permutacion(mensaje2, clave, n):
    mensaje_permutado = []
    for lista in mensaje2:
        nueva_lista = [lista[i - 1] for i in clave]
        mensaje_permutado.append(nueva_lista)
    return mensaje_permutado

#Toma los vectores y los une en una lista de números.
def lista_de_numeros(mensaje_permutado):
    lista_final = []
    for vector in mensaje_permutado:
        elemento = vector
        lista_final.extend(vector)
    return lista_final

clave = [2, 1, 3]
mensaje = [0, 11, 6, 14, 17, 8, 19, 12, 14]
n = longitud_clave(clave)
vectores = agrupar_en_vectores(mensaje, n)
nuevos_vectores = permutacion(vectores, clave, n)
nueva_lista = lista_de_numeros(nuevos_vectores)
print(nueva_lista)
```

A III: Código 16 Cálculo permutación inversa

```
#La siguiente función calcula la permutación inversa, colocando cada número en su valor correspondiente.
def clave_inversa(clave):
    clave_inversa = [0] * len(clave)
    for i, num in enumerate(clave):
        clave_inversa[num - 1] = i + 1
    return clave_inversa

clave = [4, 3, 1, 2]
inversa = clave_inversa(clave)
print("Permutación original:", clave)
print("Permutación inversa:", inversa)
```

A III: Código 17 Asociación 2

```
import string

#La siguiente función crea un diccionario para asociar cada letra del alfabeto a un número del alfabeto ASCII:
def asociacion_ASCII():
    alfabeto_mayusc = string.ascii_uppercase
    asociacion2 = {}
    for indice, letra in enumerate(alfabeto_mayusc):
        asociacion2[letra] = indice + 65
    return asociacion2

#La siguiente función itera a través de cada caracter del texto y añade su número a 'texto_numeros'.
def convertir_texto_numeros(texto, asociacion2):
    texto_numeros = ""
    for letra in texto:
        if letra.isalpha() and letra.isupper():
            numero = asociacion2[letra]
            texto_numeros += str(numero)
        else:
            raise ValueError("El texto debe ser en mayúsculas y sin espacios")
    return int(texto_numeros)

try:
    texto_original = "OK"
    asociacion_abecedario = asociacion_ASCII()
    texto_numeros = convertir_texto_numeros(texto_original, asociacion_abecedario)
    print("Texto Original:", texto_original)
    print("Equivalente numérico:", texto_numeros)

except ValueError as e:
    print(e)
```

A III: Código 18 Selección de primos p y q

```
import random

#Esta función genera pares de números primos aleatorios cuyo valor está comprendido entre 2 y 1000.
def generar_pares_primos(lista):
    dosprimos = random.sample(lista,2)
    return dosprimos

#Esta función devuelve p y q, y calcula N y xN.
def calculo_variables(par_primos):
    p = par_primos[0]
    q = par_primos[1]
    N = p*q
    xN = (p-1)*(q-1)
    return p, q, N, xN

lista_primos = [2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89, 97, 101, 103, 107, 109, 113, 127, 131, 137, 139, 149, 151, 157, 163, 167, 173, 179, 181, 191, 193, 197, 199, 211, 223, 227, 229, 233, 239, 241, 251, 257, 263, 269, 271, 277, 281, 283, 293, 307, 311, 313, 317, 331, 337, 347, 349, 353, 359, 367, 373, 379, 383, 389, 397, 401, 409, 419, 421, 431, 433, 439, 443, 449, 457, 461, 463, 467, 479, 487, 491, 499, 503, 509, 521, 523, 541, 547, 557, 563, 569, 571, 577, 587, 593, 599, 601, 607, 613, 617, 619, 631, 641, 643, 647, 653, 659, 661, 673, 677, 683, 691, 701, 709, 719, 727, 733, 739, 743, 751, 757, 761, 769, 773, 787, 797, 809, 811, 821, 823, 827, 829, 839, 853, 857, 859, 863, 877, 881, 883, 887, 907, 911, 919, 929, 937, 941, 947, 953, 967, 971, 977, 983, 991, 997]
par_primos = generar_pares_primos(lista_primos)
variables = calculo_variables(par_primos)
print("p:",variables[0],"q:",variables[1],"N:",variables[2],"xN:",variables[3])
```

A III: Código 19 Cifrado RSA

```
def Cifra_RSA(M,e,N):
    C = (M^(e))%N
    return C

M = 7975
e = 251
N = 655009
C = Cifra_RSA(M,e,N)
print("Cadena_original:", M)
print("Cadena encriptada:", C)
```

A III: Código 20 Cálculo de la clave privada d

```
#El siguiente código realiza las iteraciones del algoritmo de Euclides extendido hasta que b = 0, en ese caso devuelve el valor a que es el MCD, el de u y el de v.
def algoritmo_Euclides_extendido(a, b):

    u0 = 1
    v0 = 0
    u1 = 0
    v1 = 1

    while b != 0:
        q = a//b
        r = a - b * q
        u = u0 - q * u1
        v = v0 - q * v1
        a = b
        b = r
        u0 = u1
        v0 = v1
        u1 = u
        v1 = v
    return u0

def d(u0, b):
    d = u0 % b
    return d

A = 251
B = 653380
resultado = algoritmo_Euclides_extendido(A, B)
d = d(resultado, B)
print ("d:", d)
```

A III: Código 21 Descifrado RSA

```
def descifrado_RSA(C,d,N):
    M = (C^(d))%N
    return M

C = 313705
d = 411291
N = 655009
M = descifrado_RSA(C,d,N)
print("Cadena encriptada:", C)
print("Cadena_original:", M)
```

A III: Código 22 Asociación 2 inversa

```
import string

#Esta función crea el mismo diccionario que en el primer caso, pero en dirección inversa.
def asociacion_ASCII_inv():
    alfabeto_mayusc = string.ascii_uppercase
    asociacion2 = {}
    for indice, letra in enumerate(alfabeto_mayusc):
        asociacion2[indice + 65] = letra
    return asociacion2

#Esta función agrupa los números por parejas.
def agrupar(cadena_de_numeros):
    numeros = str(cadena_de_numeros[0])
    parejas = [int(numeros[i:i+2]) for i in range(0, len(numeros), 2)]
    return parejas

#Esta función asocia la letra correspondiente a cada número de la lista.
def convertir_numeros_texto(numeros, asociacion1):
    lista_letras = []
    for numero in numeros:
        letra = asociacion1[numero]
        lista_letras.append(letra)
    return ''.join(lista_letras)

asociacion_abecedario_inversa = asociacion_ASCII_inv()
cadena_de_numeros = [7975]
lista_numeros = agrupar(cadena_de_numeros)
texto_asociado = convertir_numeros_texto(lista_numeros, asociacion_abecedario_inversa)
print("Cadena de números:", cadena_de_numeros)
print("texto asociado:", texto_asociado)
```

A III: Código 23 Firma Digital

```
def descifrado_RSA(C,d,N):
    M = (C^(d))%N
    return M

C = 65
d = 411291
N = 655009
M = descifrado_RSA(C,d,N)
print("Cadena encriptada:", C)
print("Cadena_original:", M)
```

A III: Código 24 Firma digital comprobación

```
def Cifra_RSA(M,e,N):  
    C = (M^(e))%N  
    return C  
  
M = 496484  
e = 251  
N = 655009  
C = Cifra_RSA(M,e,N)  
print("Cadena_original:", M)  
print("Cadena encriptada:", C)
```