Dr. Kiss Gábor

Kriptográfiai alapismeretek I.

Ókortól a II. Világháború végéig

2025

Előszó

Napjainkban egyre több helyről halljuk azt, hogy hackerek titkosították egy cég adatait és csak egy jelentős összeg megfizetése után hajlandóak a titkosítást feloldani. Ilyen esetekben természetesen a rendszeres biztonsági mentés segít az adatok visszatöltésében, de ha nincs biztonsági mentés, vagy régebben készült, akkor bizony több cég megfontolja a fizetést mérlegelve a hiányzó adatok pótlásának erőforrásigényét. A fizetés sem mindig garantálja a titkosított adatok visszaszerzését, ugyanis nem garantálja semmi, hogy a pénz megérkezése után a károkozók megadják a visszafejtéshez szükséges információt. Emellett nem biztos, hogy a hackerek megfelelően leteszteltelték a programjukat és valóban a visszafejtéshez megfelelő kulcsot kapjuk, ahogy ez a történelemben már előfordult.

A tankönyv egy két részes sorozat első kötete. Ebben a kötetben az Ókortól kezdve a II. Világháború végéig használt kriptográfiai módszeretek tárgyaljuk, a második kötetben pedig a II. Világháború utáni időszakot tekintjük át napjainkig.

A könyvsorozat kettős céllal készül.

Elsősorban azok számára, akik információbiztonsággal kapcsolatos képzésen vesznek részt. Számukra elengedhetetlen, hogy kriptográfiai ismeretekkel rendelkezzenek.

Másodsorban a titkosítás iránt érdeklődők számára készült, akik szeretnének ezen a területen alapismereteket szerezni.

A tankönyv első kötete foglalkozik egy másik területtel, a szteganográfiával, mely az adatok elrejtésével foglalkozik, azok titkosítása helyett. A szteganográfia hátránya, hogy az elrejtett adat, mely megfelelő tudás birtokában információvá válik, rögtön elolvasható, ha az illető megtalálta a módját, hogyan rejtették azt el. A szteganográfia területét szintén az ókori kezdetektől mutatjuk be, de ott nem állunk meg a II. Világháború végénél, hanem eljutunk egészen a napjainkig használt technikák bemutatásáig.

A tankönyvsorozat kettős céllal készült. Egyik részről azért, mert a téma iránt érdeklődőek számára érdekes lehet a történelmi kitekintés, illetve az alkalmazott módszerek hatása a történelem alakulására. Másrészről a több, mint 20 éves oktatási tapasztalatom ezen a területen azt mutatja, hogy a napjaink titkosítási módszereinek elsajátítása könnyebben megy a korábbi módszerek ismeretében, mert részben azokhoz hasonló elemek találhatóak meg bennük, illetve addigra olyan gondolkodásmód fejlődik ki az olvasóban ezen a téren, hogy a komolyabb matematikai műveletek megértése sem lesz már akadálya az elsajátításnak.

Még egy plusz hozadéka van a könyv olvasásának, megváltozik az ember szemlélte és képes lesz az információbiztonság szemüvegén keresztül szemlélni a környezetében történő eseményeket.

A tankönyv, alapul veszi Simon Singh: Kódkönyv - A rejtjelezés és rejtjelfejtés története, 2007-ben megjelent könyvét [1].

A BMP és WAV fájlokba történő adatrejtést bemutató, saját fejlesztésű programok elkészítését Virasztó Tamás: Titkosítás és adatrejtés c. könyvének elolvasása inspirálta [2].

A Cázárkódolást, Monoalfabetikus kódolást és törését, a Vigenére kódolást szemléltető, saját fejlesztésű programok német nyelvű változatának bemutatására megtisztelő felkérést kaptam a Világ legnagyobb Számítógépmúzeumának a Heinz Nixdorf MuseumsForum igazgatójától, Norbert Ryska úrtól a Krypto&Stegano kiállítás családi napjára. Köszönetképpen David Kahn: The Codebreakers- The Comprehensive History of Secret Communication from Ancient Times to the Internet c. könyvének dedikált változatát kaptam, melyet szintén felhasználok a tankönyv elkészítése során [3].

A Cardano-rácsot bemutató program Megyesi Zoltán: Titkosírások c. könyvében leírt algoritmust követi [4].

A tankönyvsorozat bizonyos kriptográfiai eljárások kipróbálásához Python nyelvű forráskódokat is tartalmaz annak érdekében, hogy az egyes titkosítási eljárások működése könnyebben megérthető, legyen.

A tankönyvsorozat nem tér ki külön a Python nyelv bemutatására, de alapszintű programozási ismerettel nagy valószínűséggel megérthetőek így is a programsorok.

Nem törekedtünk optimális forráskód készítésére, inkább a kriptográfiai algoritmusok lépésenkénti „leprogramozását” vettük alapul.

A tankönyv I. kötete nem helyettesíti az órán való részvételt, a tananyag sikeres elsajátításának támogatása a célja a rendszeres óralátogatás mellett.

Tartalomjegyzék

Miről is lesz szó a könyvben?

A mindennapjaink során mobileszközön telefonálunk, chat-elünk, banki ügyeinket intézzük és nem feltétlenük tudatosul bennünk, hogy ezek az információáramlások jellemzően titkosítva vannak annak érdekében, hogy egy harmadik fél ne férhessen ezekhez hozzá.

Amennyiben már hallottunk róla, nem biztos, hogy tudjuk, hogyan is valósul meg a titkosítás, mennyire tekinthető biztonságosnak.

A tankönyvsorozat célja, megfelelő elméleti ismereteket nyújtani a kriptográfia területén, kitekintve a szteganográfiára is, mely az adatok elrejtésével foglalkozik.

Az első kötetben az ókortól kezdve mutatjuk be a legfontosabb adatrejtési és titkosítási módokat, ezzel is rámutatva azok történelmi jelentőségére, valamint a korábbi generációk intelligenciájára, mely lehetővé tette ezek kigondolását és használatát. Érdekes lesz az olvasó számára rácsodálkozni egy letűnt korban alaposan végigondolt titkosítási eljárására és ezzel megtanít minket kellő alázattal tekinteni elődeinkre, akikről nem feltétlenül tételeztünk volna fel ekkora tudást. Úgy gondolhatják egyesek, hogy okosabbak a korábbi generációnál, de az első kötet végére az olvasásban rájövünk, hogy csak a fejlettebb technológia biztosít előnyt számunkra.

A könyvben számtalan példát mutatunk be, hogy szemléltessük az egyes eljárások működését, így segítve az elméleti ismeretek megértését és gyakorlatba történő átültetését.

Több esetben Python nyelvű forráskódot is mellékelünk, hogy számítógépen is kipróbálhatók legyenek.

A könyv által átadott ismeretek alapján mindenki képes lesz a környezetében lévő adattárolási módok biztonsági szintjének ellenőrzésére és szükség esetén annak növelésére.

További eredményként megemlíthetjük, hogy felvértezi az olvasót egy olyan képességgel, mely információbiztonsági szemüvegen keresztül láttatja velünk a Világot.

Még egy hozadéka lesz a tankönyv elolvasásának, mégpedig az, hogy megtudjuk, minden rendszer megtörhető, minden titkosítás visszafejthető. A kérdés az, mennyi idő alatt adott számítási kapacitás mellett, illetve hogy annyi idő alatt az információtartalma a védett adatnak elévül-e.

A mai kriptográfiai rendszerek jellemzően olyan matematikai alapokon nyugszanak, melyek garantálják az ismert számítási kapacitás mellett azt, hogy évek hosszú sorára van szükség a titkosított adatok visszafejtésére. Ezt a biztonságérzetet veszélyezteti a kvantumszámítógépek megjelenése, nem véletlenük kutatják az egyes országok kriptográfusai újabb módszerek kifejlesztését, mely a megugró számítási kapacitás mellett is garantálni tudják az adataink biztonságát. A következő évek megmutatják majd, mennyire jártak sikerrel.

I. Az első ismert adatvédelmi eljárások

Bevezetés

Az írásbeliség megjelenésével együtt megjelent az igény a leírt adatok védelmére. A ókorban már a veszély egy részét elhárította az, hogy nagyon kevesen tudtak egyáltalán olvasni. Tehát csak egy szűk kör ellen kellett védeni a leírt szövegeket.

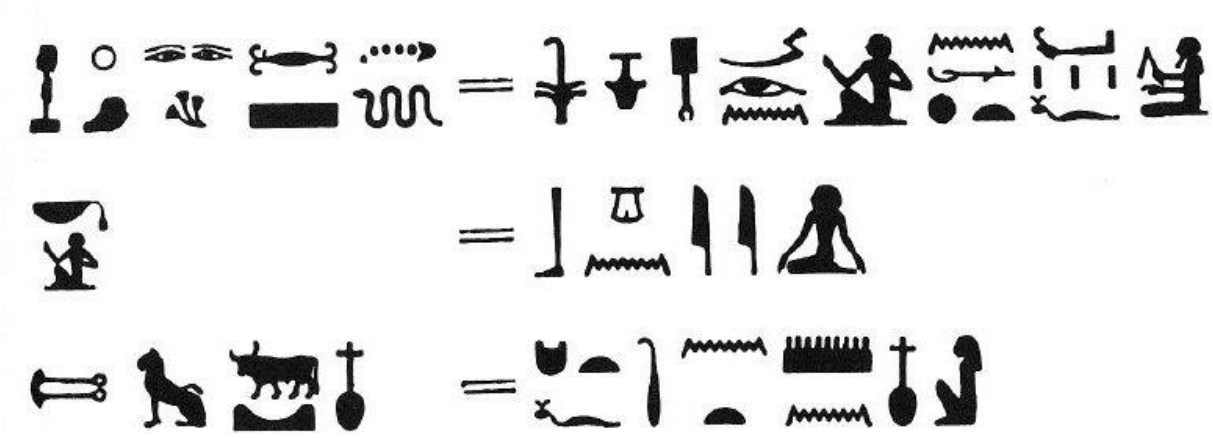
Akik tudtak olvasni, azok közül is csak kevesen férhettek hozzá, hiszen őrség által őrzött, elzárt területen tárolva egyfajta védelemmel láthatjuk el. De képzeljük el, mi van akkor, ha valaki megvesztegeti az őrséget, hogy engedjen be minket, vagy csak egyszerűen hozza ki a bizalmas iratot számunkra. Erre a lehetőségre is fel kellett készülni az ókorban élőknek, ezért különböző módszereket dolgoztak ki.

Asszíria

Az asszír, babilóniai írnokok a kevésbé ismert írásjelek használatával védték a szenzitív adatokat. Csak egy szűk csoport számára jelentéssel bíró írásjelekkel kommunikáltak egymás között. Hasonlít ahhoz, ahogy a katonai/rendőri bevetésen kézjelzésekkel kommunikálnak egymással a csoport tagjai, így némán tudnak üzenetet váltani és aki látja is őket, nem ismeri a pontos jelentését a kézjeleknek.

Egyiptom

Az egyiptomi papok hasonló módon csak a beavatottak által ismert hieroglif jelek használatával oldották meg az információvédelmet. A Dendera-i Hathor szentély falán találtak a tudósok máshol nem található hieroglif jeleket is (1. ábra).



1. ábra A Denderai Hathor szentély hieroglifái (részlet)

Kína

A kínaiak az írásjelek több jelentését kihasználva titkosították az üzeneteket. Aki ma szeretne megtanulni írni a kínai írásjelek használatával, már a tanfolyam elején szembesül azzal, hogy egy írásjelnek több jelentése is van. Ez a tény kihasználható arra, hogy csak az adott információval rendelkezők számára bújjon elő a szövegből annak igazi jelentése. Hasonlít ahhoz, amikor az emberek virágnyelven kommunikálnak egymással, vagy egy adott szó, kifejezés, mondat egy adott csoport számára a közös emlékeik miatt speciális jelentéssel bír.

Mezopotámia

Az első ismert ipari titokvédelmi eljárást ie. 1500-ból ismerjük, egy mezopotámiai tábla tartalmaz egy rejtjelezett leírást az agyagedény beüvegezésére. Mivel a beüvegezett agyagedény jóval drágábban értékesíthető volt a csillogásának köszönhetően, ezért fontos, hogy csak egy szűk kör ismeri azt az eljárást, amivel ez megvalósítható.

Napjainkban még hangsúlyosabban jelenik meg az ipari titkok védelme, ezért egy beszállítócégnek komoly feltételeknek kell megfelelnie, hogy egyáltalán a gyártó megbízza bizonyos alkatrészek készítésével.

Az ipari kémkedés ma már egy külön szakma, mely felhasználja a Social Engineering technikát, mely a megtévesztésen, pszichológiai manipuláción alapul. Előfordul, hogy cégek szakembereket bíznak meg azzal, hogy próbáljanak pl. adott irodának adott asztalának fiókjából adott iratot kijuttatni, ezzel ellenőrzik a felkészültségüket az ilyen célú támadások ellen.

II. Steganográfia

Bevezetés

Ebben a fejezetben az adatok elrejtését célzó szteganográfiával foglalkozunk.

A szteganográfia nem használ titkosítási eljárást, egyszerűen csak el akarja rejteni az üzenetet az avatatlan tekintetek elől.

Ebből következik a hátránya is, ugyanis ha megtalálják az elrejtett üzenetet, akkor el is tudják azt olvasni.

Ókor

Kína

Kínában a titkos üzenetet pl. vékony selyemszalagra írták, golyóvá gyúrták és bevonták viasszal. A golyót a futár lenyelte, vagy más módon elrejtette, hasonlóan ahogy manapság a csempészendő droggal teszik bizonyos esetekben, a célállomáson pedig az üzenet elolvashatóvá válik, amint azt sikerül a futárnak átadnia.

Az ókori görögök egy másik módszerhez folyamodtak. Egy rabszolgának leborotválták a haját, a fejbőrére tetoválták az üzenetet s megvárták míg kinő a haja. Ez után küldték el az „üzenetet" saját lábán a címzettnek. Ezzel a módszerrel ie. 500-ban Hisztiaiosz küldött így üzenetet a Perzsa birodalomból, annak felkészületlenségéről egy esetleges kis-ázsiai ión felkeléssel szemben Arisztagorasznak. Ennek a módszernek az alkalmazásához némi időre van szükség. A mai felgyorsul társadalmunkban az információáramlásnak ez a sebessége nem lenne elfogadható a felek számára.

Görög városállamok

Ie. 480-ban Xerxész zokon vette, hogy Athén és Spárta nem fizeti be az adót. Az adóbehajtás indokával igyekezett meghódítani a görög területeket. Összehívta a történelem addigi legnagyobb haderejét és a görögök meglepetésszerű lerohanására készült. Demaratosz, aki látta a perzsa had készülődését (ie. 486) figyelmezteti a görögöket a támadására egy írótábla segítségével. Az írótábla modern változatama is létezik, de papírt rögzítenek egy kapoccsal a tetején, a tábla alátámasztást ad az íráshoz. Az ókorban, amikor még nem volt papír, az írótáblát úgy kell elképzelni, mint egy összehajtható két falapot. A lapoknak van egy alacsony kerete. A falapok belső felére viaszréteget vittek fel (a keret arra szolgált, hogy ne folyjon ki oldalt a viasz, amikor az felviszik a keret által határolt mélyebb részre. A viaszra írták a szöveget és lepecsételve hitelesítették. A két falapot összecsukva adott védelmet a külső oldal a belső rétegre írt szövegnek. Korábban ezért volt a pecsétgyűrűnek jelentősége, hiszen jellemzően a tulajdonosához tartozott, így az adott pecsétlenyomat igazolta a küldő/aláíró felet. Ma az aláírásunkkal és tanúkkal igazoljuk egy papír alapú dokumentum hitelességét. De térjünk vissza Demaratoszhoz. Lekaparta a viaszt a tábláról és falapra kaparta az üzenetet, majd azt viasszal bevonta, mintha üres lenne. Ezt a viasztáblát küldte el Spártába Leonidaszhoz. Mivel ebben az időben természetes volt az üres írótáblák „utazása", hasonlóan ahhoz, hogy valakinek adunk papírt, hogy tudjon jegyzetelni az órán, mert nem hozott magával. Leonidasz és társaihoz az üresnek látszó írótábla az ellenőrzési pontokon áthaladva eljutott, de nem tudták mire vélni. Leonidasz felesége, Gorgó javaslatára nézték meg, mi lehet a viaszréteg alatt, így szerezve tudomást a perzsa sereg készülődéséről. Az üzenet hatására a görögök flottát építettek és Szalamisznál megverték a perzsa hajóhadat. Ugyan a spártaiak nem tudták feltartóztatni a

perzsákat a Thermopülai-szorosnál, de Xerxész elveszítve az utánpótlást, kénytelen volt visszavonulni. A további történéseket már ismerjük a történelemkönyvekből.

Polybios

Polybios (Kr.e. 200 – 120. élt görög történetíró, később még megjelenik a könyvünkben) a háború tudományáról írt könyvében titkosírási módszerek mellett rejtési módszereket is leír.

Az általa ismertetett módszer azt taglalja, hogy egy ártatlannak látszó szöveg majdnem észrevehetetlenül megjelölt betűit összeolvasva kapjuk meg a rejtett üzenetet.

Javaslata szerint egy ólomlemezre írt üzenet egyes betűit apró tűszúrásokkal kell megjelölni, magát az ólomlemezt belevarrni a saru talpába. A futár ebben a saruban viszi el az üzenetet, ha átkutatják, nem találnak nála semmi érdemi tárgyat. A címzettnél a saruból kibontva a lemezt az fény felé tartva jól láthatóan előtűntek a jelölések és a valódi üzenet a megjelölt betűk összeolvasásával előáll.

Középkor

Láthatatlan tinta

Láthatatlan tinta használatára már az első évszázadban van utalás, miszerint pitypang tejével írhatunk papiruszra. Melegítés hatására válik láthatóvá az ilyen módon írt szöveg a benne lévő széntartalom elszíneződésének köszönhetően.

A történelem során az is előfordult, hogy a láthatatlan tinta hiányában a kémek a saját vizeletükkel írtak. A széntartalom itt is érdekes az olvashatóság szempontjából.

A technikát a legújabb korban is használják, pl. Lenin citromlével írta a bizalmas leveleit.

A citromlé kapcsán álljon itt McArthur Wheeler története, aki olvasott a citromlé láthatatlan tintaként való alkalmazásának lehetőségéről. 1995-ben két bankot is kirabolt, de a rendőrség tudta azonosítani a biztonsági kamerák képei alapján. McArthur Wheeler meglepődott, amikor a rendőrség az otthonában felkereste, mert úgy gondolta, hogyha a arcát bekeni citromlével, akkor a kamerák számára azonosíthatatlan lesz. A történelem nem igazolta ezt a feltevését.

A XVI. Században egy olasz tudós leírta, hogyan lehet egy titkos üzenetet keménytojás segítségével továbbítani. Timsó és ecet elegyéből lehet olyan „tintát” készíteni, amivel a tojáshéjra írva az a pórusokon áthatol, az írás pedig kirajzolódik a fehérjén, mely a héj eltávolítása után láthatóvá válik. Láthatjuk, hogy ezzel a módszerrel hosszú üzenet továbbításához több tojásra van szükség és érdekes lesz azok sorrendje.

II. Világháború

Polybios korábban említett módszeréhez hasonlót használtak a második világháborúban úgy, hogy a diplomáciai postával küldött nyomtatott újságok egyes betűit jelölték meg pl. szabad szemmel nem is látható mikropontokkal.

Egy másik módszert is kidolgoztak, amelyhez szükség volt a fényképészeti eljárások fejlődésére. A Dél-Amerikában tevékenykedő német ügynökök fényképészeti eljárással egy gépelt oldalnyi szöveget egy milliméternél is kisebb pöttyé zsugorítottak több menetben és egy látszólag semmitmondó levél valamelyik mondata után helyezték el a kivágott filmdarabot. Érdekesség, hogy egészen addig nem tűnt fel az angoloknak ez a módszer, míg valaki észre nem vette, hogy az egyik ilyen mondatvégi ponton megcsillan a nap fénye. Ekkor kinagyították a filmdarabot és olvashatóvá vált az elrejtett szöveg.

Modern szteganográfia

Természetesen napjainkban is használhatjuk az adatrejtés technikáját, csak alkalmazkodnunk kell a technológiai újításokhoz.

A szteganográfiával objektumokba (E-mail, kép, audiofájl) rejtett üzenetek, olyan dolgokat tartalmazhatnak, amelyek segítségével bármilyen államra, illetve polgárokra veszélyes terrorista akció végrehajthatóvá válik. pl. egy város épületeinek pontos tervrajza, bomba elhelyezésének helye, stb.

Szövegfájlok

Egy egyszerű TXT fájlt is felhasználhatunk adatrejtésre. Az elrejtendő betűt ASCII kódjának megfelelő bitsorozatot használva megfelelő mennyiségű szóközt szúrjunk a sor végére (0/1). Így 8 sorral 1 betűnek az ASCII kódját tudjuk elrejteni.

Egy másik módszer, hogy egy MS Word dokumentumban az összeolvasandó betűket 1 ponttal nagyobb/kisebb méretben írjuk, más betűtípussal írjuk, vagy az összeolvasandó szavak előtti szóköz méretét állítjuk kisebbre, vagy nagyobbra.

Lehetőségünk van képekbe, hangfájlokba való adatrejtésre is.

A média fájl lehet tömörített, vagy tömörítetlen, a módszer azonos, maximum a megvalósítás eltérő.

LSB technika

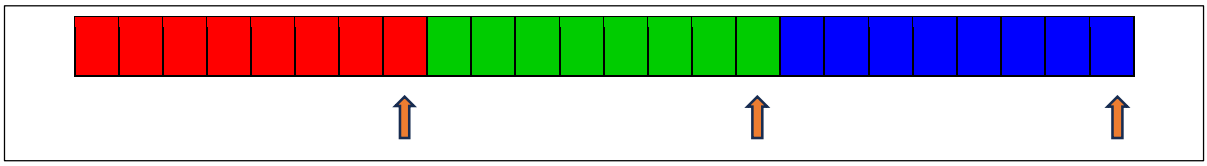
A módszer az LSB (Least Significant Bit) technikán alapul. Az alkalmazásánál a legkisebb helyiértékű bitet egy adatbájtban kicseréljük egy számunkra hasznos bitre. Az elrejtendő karaktert először bitjeire kell bontani, majd ezeket a biteket egyesével kell elrejteni

Az elkészített, saját fejlesztésű program 24 bites BMP képfájlokat használ, illetve WAV formátumú hangfájlokat. Mivel ezek nem tömörítettek, ezért a 24 bites (True Color) BMF fájlban 3 színbájt adja meg egy pixel színét, rendre a vörös(R), zöld (G) és a kék (B) színösszetevőkkel.

A WAV fájlok esetében magas, illetve alacsony hangokat reprezentál 1-1 bájt, illetve sztereó fájlok esetén mindkét oldalra vonatkozóan.

A képfájlok esetében minden színösszetevő a maga 256 árnyalatával járul hozzá az adott pixel színéhez, így egy pixel $2^{(8*3)}$=16.777.216 különböző színt vehet fel. A színkomponensek egymás után folytonosan helyezkednek el a BMP fájlban és tulajdonképpen ezeknek a száma határozza meg a fájl méretét is. A pixelek kirajzolása a kép bal alsó sarkától kezdődik, majd egy vízszintes sor kirajzolása után egy sorral feljebb folytatódik egészen addig, amíg a kirajzolás el nem éri a jobb felső sarkot.

Az LSB technika lényege képekre alkalmazva, hogy a legkisebb helyiértékű színbitet kicseréljük egy számunkra hasznos bitre, hiszen ennek megváltozása emberi szemmel érzékelhetetlen (2. ábra). Egy alfanumerikus karaktert így el tudunk rejteni 8 bájt színinformációban.



2. ábra Az LSB technika színbájtok esetében

Nézzünk egy példát:

az elrejtendő karakter pl. az A betű (01000001), melynek az ASCII kódjának megfelelő nyolc bites sorozatot rejtjük el a színbájtokba. Az elrejtéséhez 8 db. szítbájtra lesz szükség (1. táblázat). A táblázat első felében a kiválasztott 8 színbájt értékeit mutatjuk meg kettes számrendszerben, az alsó felében pedig az elrejtendő információnak („A" betű) a bitértékeinek beírása után a módosított értékeket (vastag és dőlt betűvel).

1. táblázat LSB technika alkalmazása (példa)

Az eredeti színbájtok értéke:	R	G	B	R
	00100111	11101001	11001000	00100111
	G	B	R	G
	11001000	11101001	11001000	10001010
A megváltozott színbájtok értéke:	R	G	B	R
	00100111	11101001	11001000	00100111
	G	B	R	G
	11001000	11101001	11001000	10001010

A táblázatban látható, hogy a legkisebb helyiértékű bit felhasználásakor 50% az esély arra, hogy egyáltalán módosítást kell végrehajtanunk, mivel ha eleve 0 áll ott, ahova 0-át, vagy 1-es áll ott, ahova 1-est kellene írnunk, nem kell módosítanunk. Abban az esetben, amikor módosítunk, az adott szín színskáláján csak 1 értékkel módosítunk, ezáltal a változás az emberi szem számára változatlan marad. Egy szöveg esetén minden egyes betűt ezzel a módszerrel tudjuk elrejteni a képben.

Kiolvasáshoz az egyes színbájtok legkisebb helyiértékű bitjeit 8-as csoportokba rakjuk, visszakapjuk a betű ASCII kódját kettes számrendszerbe és visszakonvertáljuk karakterekké.

Könnyen belátható, hogy fájl mérete sem változik az információ elrejtése után, hiszen csak 1-es írunk át 0-ra, vagy 0-át 1-esre. A 24 bites képekbe rejthető információ maximális mérete függ a kép felbontásától, mivel a színbájtok száma pontosan a kép vízszintes- és függőleges felbontásának szorzatának háromszorosa. Könnyen megkaphatjuk a maximálisan elrejthető információt bájtokban, ha összeszorozzuk a kép vízszintes felbontását a függőleges felbontásával, ezt hárommal (színkomponensek: R,G,B) beszorozzuk és osztjuk nyolccal, mivel minden 8 bitből csak 1 bitet használtunk fel információ elrejtése céljából.

Egy 1200x800 pixeles 24 bites kép esetén, mivel minden egyes pixelt 3 színbájt ír le, az alábbi számítás adja meg az elrejthető karakterek számát:

$$\frac{1200x800x3}{8} = 360.000\ karakter$$

Belátható, hogy lehetőségünk van arra is, hogy ne csak a legutolsó bitet módosítsuk, hanem kettőt, vagy akár négyet is (mind a 8-at nem érdemes, mert azonnal észrevehető a változás).

A két színbit megváltoztatásával a képletünk az alábbiak szerint változik, egy betű elrejtéséhez ugyanis már csak 4 színbájtra van szükség (2. táblázat):

$$\frac{1200x800x3}{4} = 720.000\ karakter$$

2. táblázat LSB technika alkalmazása 2 bit felhasználásával (példa)

Eredeti szín-byte	Érték	A betű ASCII kódja (01000001)	Módosított szín-byte	Érték
00100111	39	01	00100<u>1</u>01	37
11101001	233	00	1110100<u>0</u>	232
11001000	200	00	11001000	200
00100111	39	01	00100<u>1</u>01	37

Felhasználva négy színbitet a rejtéshez a képletünk az alábbiak szerint változik, egy betű elrejtéséhez ugyanis már csak 2 színbájtra van szükség (3. táblázat):

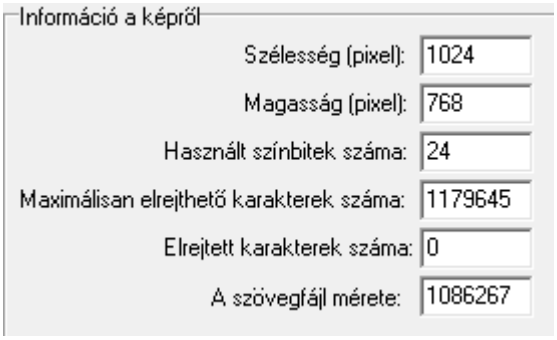
$$\frac{1200x800x3}{2} = 1.440.000\ karakter$$

3. táblázat LSB technika alkalmazása 2 bit felhasználásával (példa)

Eredeti szín-byte	Érték	A betű ASCII kódja (01000001)	Módosított szín-byte	Érték
00100111	39	0100	00100<u>00</u>	36
11101001	233	0001	1110<u>0</u>001	225

Megjegyzendő, hogy az Új Testamentum angol nyelvű változata a 1.086.267 karakter, tehát elfér egy 1200x800 pixeles képben 4 színbájt felhasználásával!

Most nézzük meg az LSB technikát bemutató programot. Amikor megnyitunk egy fájlt, akkor érdemes lekérdezni a méretét (3. ábra).



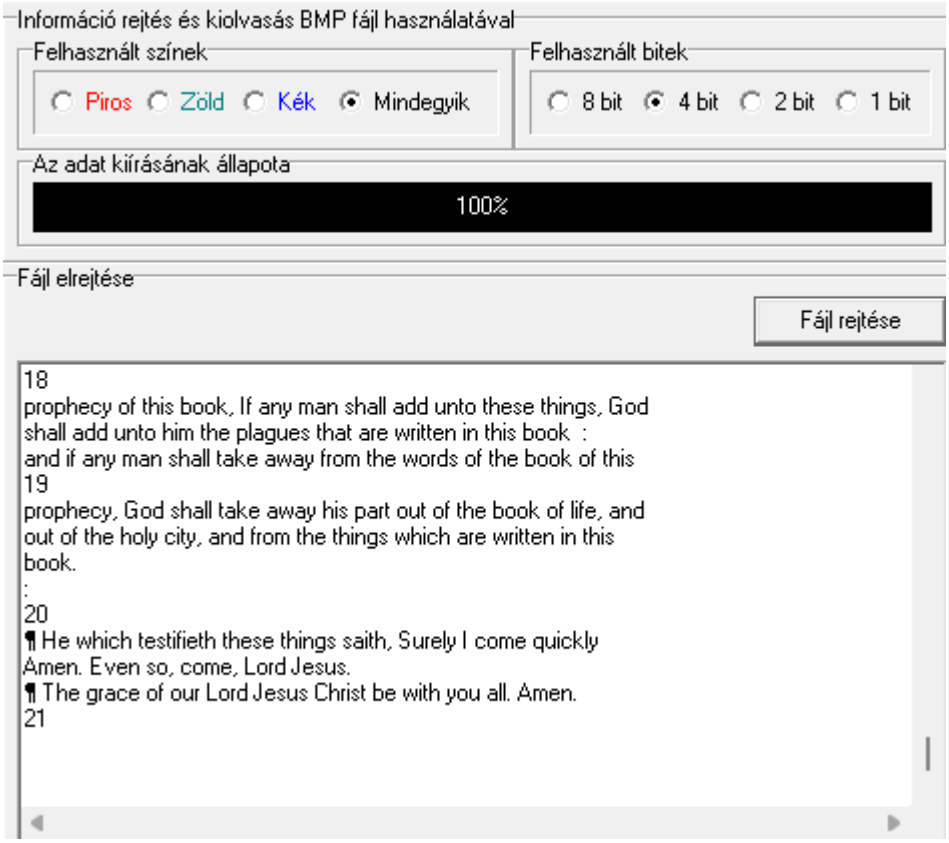
3. ábra Az LSB technika fájl információk

Az elrejtéshez meg kell adnunk, hogy melyik színt leíró bájtokat használjuk fel az elrejtésre (lehet kár csak a pirosba rejteni, de akkor az elrejthető karakterek száma csak a harmada lesz, mintha minden szint felhasználnánk).

Ezen felül azt is megadhatjuk, hogy hány bitet szeretnénk felhasználni az adatrejtéshez (4. ábra).

A 3. ábra alsó részében látható, hogy a Új Testamentum mérete több, mint 1.000.000 karakter, így ahhoz, hogy elrejtsem, 4 bitet kell felhasználnom.

A 4. ábrán látható a sokeres elrejtés, valamint az elrejtett fájl vége, hogy lássuk, mit rejtett el.



4. ábra Az LSB elrejtési tulajdonságok

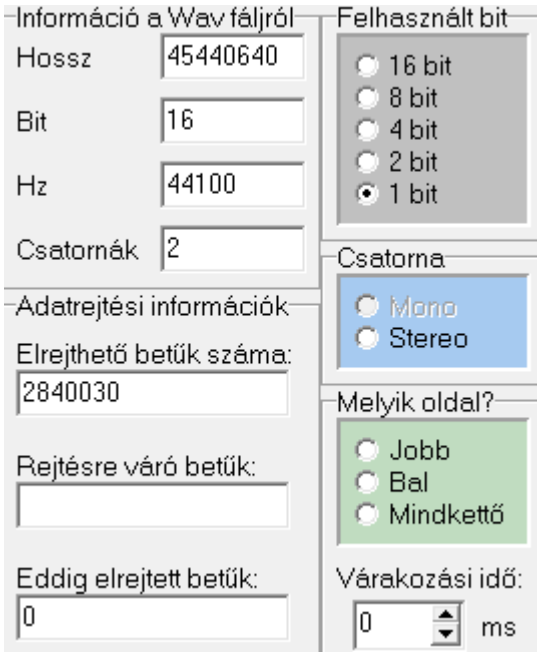
Az 5. ábrán látható képet használjuk fel a teljes Új Testamentum szövegének az elrejtésére.

A sikeres elrejtés után megnyitva a módosított képet (6. ábra), látjuk, hogy a 4 bit felhasználása már ilyen kagylósodást mutató változást okoz azokon a helyeken, ahol közel azonos színárnyalatú részek vannak nagyobb felületen. Emiatt, ha nem akarunk lebukni ezzel a technikával, vagy nagyobb felbontású képet, vagy pedig olyan képet választunk, ahol nincs pl. kék égbolt a két tetején, ahol ez a kagylósodás megjelenne és felismerhetővé tenné a kép megváltozását.

5. ábra Az eredeti kép

6. ábra A 4 bitben módosított kép

Hangfájlok esetében azonos az LSB technika használata (7. ábra), csak az a probléma, hogy az emberi fület nehezebb becsapni, mint az emberi szemet, ezért már egy bitet felhasználó rejtésnél is zaj hallgató a hangfájlban az eredeti tartalom mellett, kb. olyan hatást keltve, mintha régi felvétel lenne, miközben az eredetiben nem hallunk ilyesmit.



7. ábra LSB technika lehetőségei WAV fájlok esetén

Amikor az LSB technikát alkalmazzák üzenet elrejtésére, az Internetről letöltenek egy megfelelő képet, módosítják a fentiek alapján és elküldik a címzettnek. Aki figyeli a hálózati forgalmunkat, azt látja, hogy lementettünk egy képet és kis idő múlva továbbítottuk egy levélben. Mivel a fájl mérete nem változik meg, megnézve a kép képet pedig nem fedezünk fel változást, a rejtett üzenetküldésünk titokban marad (a technikát alkalmazták a Ikertornyok elleni támadás (2001.09.11) megszervezésében).

Az LSB technika ellenállása a különböző behatásokkal szemben

Egy egyszerű képszerkesztő programmal is könnyen kárt tudnak tenni a rejtett adatokat tartalmazó képeinkben, ha a címzetthez való utazása során valaki elfogja és módosítva küldi tovább a célállomás felé. A képek bizonyos területeinek átszínezésekor, szerkesztésekor a színekben tárolt információink megváltozhatnak, és használhatatlanokká válhatnak. Akár 1 bit megváltozása esetén is, a bit helyétől függően teljesen más-más ASCII karaktert olvas ki a program a képből érthetetlenné téve az elrejtett szöveget a kiolvasáskor.

Rejtésnél a kép alsóbb régióit használjuk a leggyakrabban (onnan rajzolódik ki a kép, tehát az első bájtok (az információs fejrész után) az alsó pixelek színét írják le), tehát ha szerencsénk van, akkor a képszerkesztés során csak a felső rész változtatják meg és a képbe rejtett adataink így sérülésmentesek maradnak. Ha egy képen belül több helyre is elrejtjük ugyanazokat az adatokat, akkor nagyobb az esélyünk, hogy ezek a területek megőrzik eredeti értéküket és később is használható információval szolgálhatnak. Ekkor természetesen rövidebb szöveget használunk, hogy többször be lehessen rejteni a képbe.

III. Kriptográfia

Bevezetés

A szteganográfiánál láttuk, hogy a mai napig használható technika, de van egy alapvető gyengesége: amennyiben az üzenetet megtalálják, akkor tartalmára azonnal fényderűl.

Ennek okáért a szteganográfiával párhuzamosan kialakult a kriptográfia is (a görög krüptosz (rejtett)), melynek célja nem az üzenet elrejtés, hanem tartalmának megfelelő ismeret hiányában annak elolvashatatlanná tétele.

Tehát nem elrejtjük az üzenetet, hanem titkosítjuk, kódoljuk, rejtjelezük, siffrírozzuk.

Az ilyen titkosított információ megfejtése a desiffrírozás, visszafejtés, dekódolás.

Az elkódolt üzenetet akár egy plakátra is kitehetjük, aki nem ismeri a módszert, amivel eltitkosítottuk, nem fogja tudni azt visszafejteni (később megtárgyaljuk, miért mégis).

A kriptográfia maga is két ágra oszlik: az átrendezésre (keverésre) és a behelyettesítésre.

Ókor

Betűk átrendezése

A keveréses eljárásnál a betűket egyszerűen átrendezik, megváltoztatják a sorrendjüket.

Egy nagyon rövid szövegeknél nem túlságosan megbízható, mert csak korlátozott számban lehet különböző sorrendbe állítani néhány betűt.

Három betű mindössze 6 féle sorrendben írható le: aki, aik, iak, ika, kai, kia.

A betűk számának növekedésével a lehetséges elrendezések száma exponenciálisan emelkedik és keverési módszer pontos ismerete nélkül nem állítható vissza az eredeti sorrend.

VEGYÜK PÉLDÁUL EZT A RÖVIDKE KIS MONDATOT.

Mindössze 35 betűt tartalmaz, mégis $5*10^{31}$ sorrendben lehet leírni.

A feladónak és a címzettnek meg kell állapodnia az átrendezés logikájában. Pl. az üzenet betűit váltakozva egy felső, ill. alsó sorba írják, majd az alsó sort a felső végéhez ragasztják (fésűs módszer). A címzett az eljárás egyszerű megfordításával megfejtheti az üzenetet.

Több formája létezik. Lehet több sorba írni, betűpárokat felcserélni stb.

Az egyik változatnál az üzenet páratlan betűi az első sorba, a páros betűi a második sorba kerülnek:

E T Z R T É T T O Í A I
Z S E E N M I K S T N

Majd egymás után fűzbe elküldjük az üzenetet:

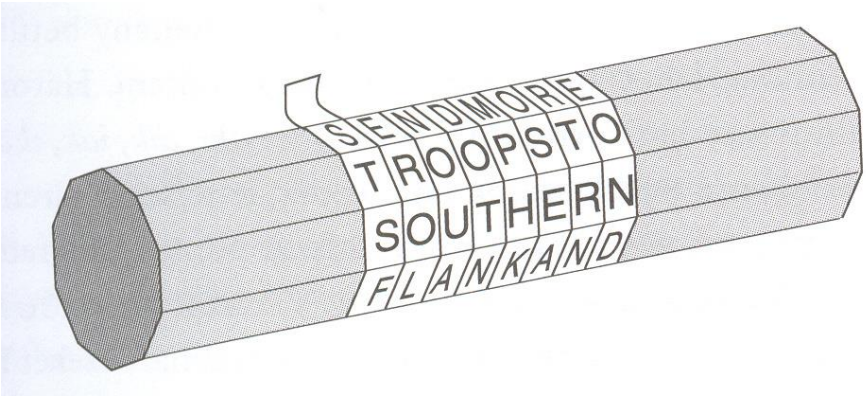
E T Z R T É T T O Í A I Z S E E N M I K S T N

A címzettnek el kell feleznie a szöveget két sorba bontva:

E T Z R T É T T O Í A I
Z S E E N M I K S T N

Majd váltakozva olvasva az első és második sor betűit visszakapja a kódolt szöveget:
E Z T S Z E R E T N É M T I T K O S Í T A N I

Az átrendezésre egy kiváló példa a világ első katonai rejtjelező eszközével a spártai szkütalé-vel kapcsolatos. Ez egy szabályos sokszög alakú (adott átmérőjű hasáb) (hadvezéri pálca), melyre egy pergamen-, vagy bőrcsíkot tekertek (8. ábra)



8. ábra Szkütalé használata [1]

A feladó a pálca hossztengelyével párhuzamos sorokban írta az üzenetet a bőrre, vagy pergamenre, majd letekerte a szalagot, mely értelmetlen betűsornak tűnik, mivel az üzenet betűi átrendeződtek. A futár ezt derékszíjnak álcázva befelé fordítva (ne legyen feltűnő a betűsorozat) viszi magával. A címzett egyszerűen rátekeri a szíjat egy azonos átmérőjű és oldalszámú rúdra és el tudja olvasni az eredeti üzenetet. Érdemes végiggondolni, hogy a használatához egy standardizált megmunkálásra van szükség mindegyik szkütalé elkészítésekor.

Ie. 404-ben Lüszandrosz spártai hadvezér ezt a módszert alkalmazva kapott értesítést egy Perzsa támadásról, melyre időben fel tudott készülni és vissza tudta verni a támadást.

Diákkoromban a 4. osztályban (1980. ☹) magyar órán a szöveggyűjteményből olvastuk az alábbi szöveget:

Gárdonyi Géza: Egy magyar rab levele (részlet)

„*Kedves ezüstös drága dádém!*

Ezer nemes arany tizedét örömmel ropogtasd örök keserűség keservét ivó magzatodért! Egészségem gyöngy. A vaj árt. Ritkán óhajtom sóval, borssal.
Ócska lepedőben szárítkozom álmomban zivataros estén.
Matyi bátyám egypár rózsát, rezet, ezüstöt, libát egy lapos leveleddel eressze hajlékomba!
Erzsi tűt, faggyút, ollót, gombot, levendulát adj!

Laci, nefelejts! Imre.”

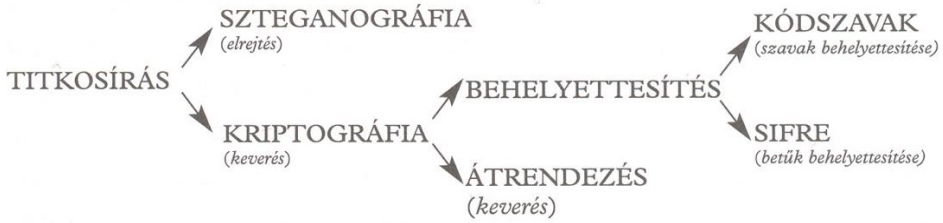
Az üzenet furcsának tűnhet, de a szavainak első betűit összeolvasva kapunk egy újabb szöveget:
„Kedden a török kimegy a városból száz emberrel el lehet foglalni”

A titkosítási algoritmus azt kívánja meg, hogy az üzenet kódolásához olyan szavakat fűzzek egymás után, melyek kezdő betűi megfelelnek az elküldendő üzenet egyes betűinek.

Amivel tudjuk csökkenteni a lebukás veszélyét az az, ha az így előálló szövegnek értelme is lesz. A módszer tovább fejleszthető, ha rendre az első, majd a következő szóban a második, stb. betűk az eredeti üzenet betűi.

Monoalfabetikus kódolás

Az átrendezés alternatívája a behelyettesítés, amikor egy betűt egy másik betűvel, vagy szimbólummal helyettesítünk (9. ábra).



9. ábra A titkosírás módjai [1]

A Kámaszutra titkosítási tétele

A behelyettesítésre az egyik legrégebbi leírást a *Kámaszutrá-ban* találjuk.

A Kámaszutra hatvannégy művészet tanulmányozását írja elő a nőknek, például a főzés, az öltözködés, az illatszerek készítése, stb. A felsorolás néhány kevésbé kézenfekvő tételt is tartalmaz: bűvészkedés, sakk, könyvkötés és ácsmesterség. A felsorolás 45. számú tétele a titkosírás művészete, amely révén titokban tarthatják viszonyaikat.

Az egyik ajánlott módszer az „ábécé” betűinek találomra történő párosítása, majd azok kölcsönös megfeleltetése. Egy példa a behelyettesítésre az angol ábécé esetében (4. táblázat):

4. táblázat Behelyettesítő kód (példa)

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

A titkosírásnak ezt a formáját *behelyettesítő* kódnak nevezik, mivel a nyílt szöveg minden betűjét egy másikkal helyettesítik. A példa szerint, ha az üzenetben „A” betűt kellene írnom, helyette az "W" betűt írom le, „W” betű helyett az „A” betűt és így tovább.

Az átrendezéses módszernél a betűk megtartják eredeti hangértéküket, de megváltoztatják a helyzetüket az üzenetben (lásd korábbi fésűs módszer).

A behelyettesítéses módszernél megmarad a helyzetük, de megváltozik a hangértékük.

Az eredmény azonos, olvashatatlanná teszi a szöveget, melyet csak bizonyos ismeret birtokában tudunk újra elolvasni.

Caesar-kódolás

A behelyettesítéses módszert Julius Caesar is használta Ciceróval történő levelezésekor a Gall-háborúk idején.

Az általa kidolgozott módszer szemben a korábban említett, rendszertelen cserélgetés helyett egy könnyebben megjegyezhető metódust használt. Caesar minden betű helyett az ábécében utána következő harmadikat írta le (ABC Caesar idejében):

```
Nyílt ábécé:       abcdefghilmnopqrstvx
Kódábécé:          DEFGHILMNOPQRSTVXABC

Nyílt szöveg:      veni ,     vidi,      vici
Kódolt szöveg:     BHQN ,     BNGN,      BNFN
```

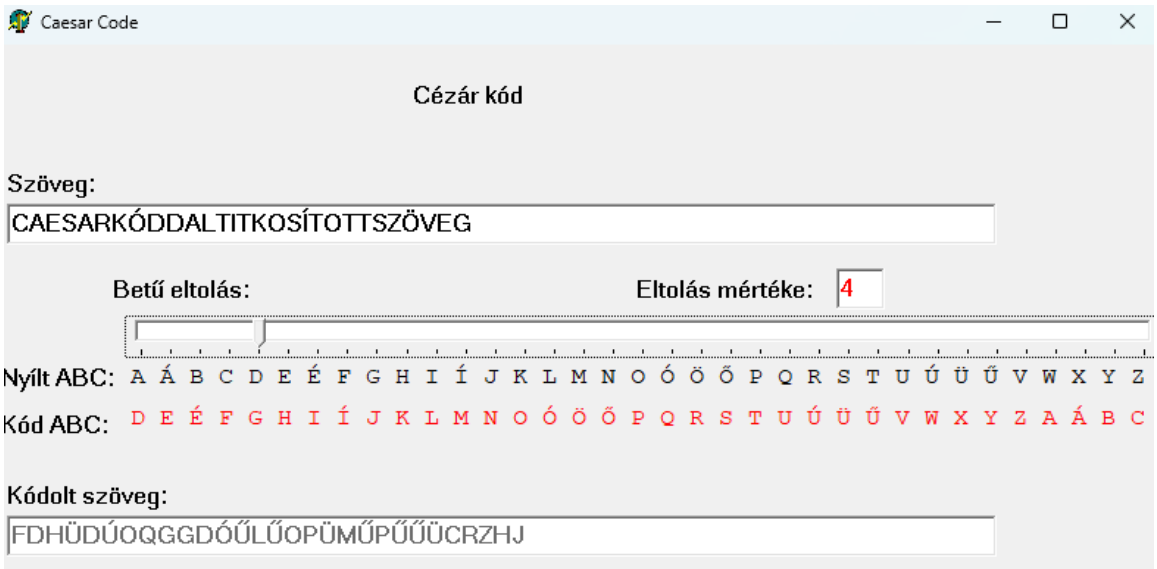
Caesar ráadásul a latin betűket görögökkel helyettesítette, tovább növelve a kódolás biztonságát.

A *nyílt ábécé* az eredeti üzenetben használt ábécét jelenti, a *kódábécé* pedig az eredeti betűket helyettesítő betűkre utalnak.

Julius Caesar utódja az első római császár Augustus Caesar a módszeren annyit változtatott, hogy nem három, hanem egy betűvel való eltolást alkalmazott és az X-et A helyett AA-val helyettesítette.

```
Nyílt ábécé:       abcdefghilmnopqrstvx
Kódábécé:          BCDEFGHILMNOPQRSTVXAA

Nyílt szöveg:      veni ,     vidi,      vici
Kódolt szöveg:     XFOL ,     XLEL,      XLDL
```



10. ábra Titkosítási algoritmus [1]

Természetesen nem csak angol, hanem pl. magyar ABC esetén is használható a Caesar-kódolás, ahogy a kifejlesztett program felülete is mutatja.

Egy csúszkával állítjuk be az eltolás mértékét és rögtön látjuk alatta az eltolt ABC-t és a kódolt szöveget (10. ábra.).

Caesar módszerének numerikus változatát Bernardo Provenzano szicíliai maffiafőnök is használta. Az 2006-ban történő elfogásáig minden betű helyett az ábécében utána következő harmadik betű sorszámát írta le.

```
Nyílt ábécé: abcdefghijklmnopqrstuvxyz
Kódábécé:    456789.............123
```

Mivel a módszer az ókorban használt Caesar-kód néven ismert kódolást használja, csak számokkal, a kriptográfiai csapat számára nem jelenthetett komoly kihívást annak megfejtése, ahogy ezt már vélhetően Bernardo Provenzano is tudja.

Caesar-kód a neve tehát minden olyan titkosírásnak, amelyben minden betűt egy másik betű vagy jel jelez egy eltolt ABC-t használva. Angol ABC esetén (26 betű), 1 és 25 között bármilyen értékű eltolás, tehát 25 különféle kódábécé generálás a lehetséges.

A Caesar kódolás egy lehetséges megoldásának Python forráskódját az 2. számú melléklet tartalmazza.

Amennyiben tudjuk, hogy az elfogott üzenet írója Caesar-kódolást egyszerűen végigpróbálja egyesével az egyes eltolásokat addig, amíg értelmes üzenetet nem kap eredményül (max. 25 eltolást). Ezt a próbálgatásos módszert nevezik a nyers erő (brute force) módszerének és minden titkosítás esetén működik (*kicsit előre szaladtunk, de itt lehet a legkönnyebben megérteni, miért is hoz eredményt mindenképp az összes lehetőség kipróbálása*)

A Caesar kódolás törését is tartalmazó algoritmus Python forráskódját az 2 .számú melléklet tartalmazza.

Amennyiben az egyszerű eltolás helyett a Kámaszutra véletlen behelyettesítéses módszerét használjuk, tehát az angol nyílt ábécé bármely elrendezését lehetővé tesszük, akkor 26!, azaz $403*10^{24}$ betűkeverés létezik, tehát ugyanennyi különböző kódábécé.

Lássuk leírva, mekkora számról is van szó:

403.000.000.000.000. 000.000. 000.000

lehetőség.

Magyar ABC-t használva (csak az egy karakterből állókat) még ezt is meghaladó értéket kapunk.

Hogy jobban érzékeljük, mekkora szám is a $403*10^{24}$, ha a brute force módszert használva szeretnénk egy általunk ismeretlen behelyettesítéses kódolást megtörni és másodpercenként tudnánk egy lehetséges kódabécét kipróbálni, hogy megtaláltuk-e a megfelelőt, akkor is nagyjából egymilliárdszor annyi ideig tartana kipróbálni az összes lehetséges változatot, mint a világegyetem élettartama.

A középkor kriptográfusai ezt nem tudták ugyan kiszámolni (a faktoriálisszámítást csak később dolgozták ki), de azt tudták, hogy annyira sok párosítás létezik, hogy a rendelkezésre álló eszközökkel (lúdtoll és pergamen) lehetetlen egy emberi életöltő alatt kipróbálni az

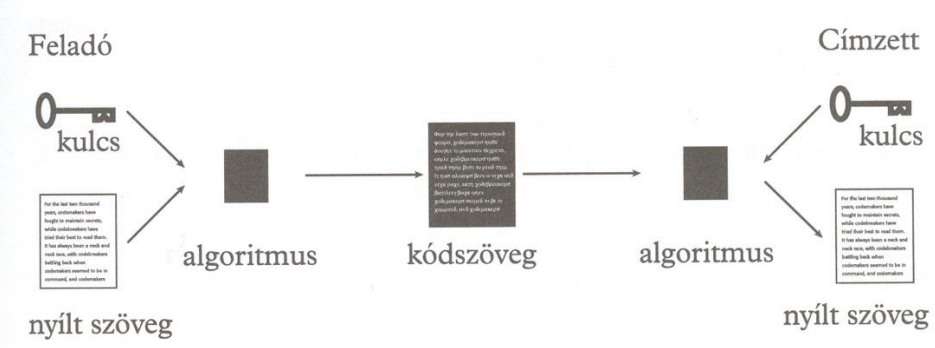
összeset, így az ezzel a módszerrel titkosított üzenet tartalma régen elévül, ha még több ezren is állnak neki az összes lehetséges betűpárosítás kipróbálásához.

Tehát látjuk, hogy a titkosítási módszer egyszerűen használható , mégis úgy tűnik, hogy elég védelmet nyújt a próbálgatásos (brute force) támadás ellen.

Ami még lehetőségként előttünk áll, hogy a karaktereket nem más karakterekkel, hanem szimbólumokkal helyettesítjük, így elijesztve a visszafejtéssel próbálkozókat (a védelmen nem változtat, ha szimbólumot használunk, csak esetleg bonyolultabbnak tűnik maga az írás miatta lásd később).

Elhangzott már a védelmi szint fogalma, mely hivatott megmutatni, hogy egy adott titkosítási algoritmus mennyire nehezen törhető. Minden egyes rejtjelezés vizsgálható az *algoritmusnak* nevezett általános, és a *kulcsnak* nevezett konkrét módszere tükrében (11.ábra).

Az fenti példában az algoritmus abból áll, hogy a nyílt szöveg minden betűjét a kódábécé valamelyik betűjével helyettesítjük, a kódábécé pedig a normál ábécé bármely elrendezése lehet.



11. ábra Titkosítási algoritmus [1]

A titkosított szöveg kulcs nélküli megfejtését kriptoanalízisnek, vagy kódfeltörésnek nevezzük.

A feltörésnek két esete van:

- vagy található valamilyen algoritmus, mely lehetővé teszi a kulcs nélküli megfejtést,
- vagy nincs hozzá ismert megfejtési algoritmus. Ebben az esetben nincs más mód a feltörésre, mint az összes lehetséges kulcs kipróbálása. Ezt nevezik a nyers erő (brute force) módszerének.

Megint kicsit előre ugorva, az első eset, amikor ismert egy backdoor-nak nevezett lehetőség pl. egy szoftver esetén, hogy hozzáférjünk annak adataihoz anélkül, hogy ismernénk a felhasználó belépési adatait. Nézzük egy példán keresztül jobban megvilágítva a backdoor fogalmát: A kollégám nemrég megkeresett, hogy van számzáras kerékpárlakatja, amit nem tud kinyitni és használni, mert elfelejtette a nyitáshoz szükséges számkombinációt. A brute force módszernél egyesével végigpróbáljuk a számokat addig, amíg ki nem nyitjuk azt. A backdoor módszer esetében ismert egy trükk, amivel a próbálgatások száma lecsökkenthető. Húzzuk a lakatot és forgatjuk lassan a legbelső számgyűrűt. Amikor nagyon picit enged a lakat, megtaláltuk a jó számot. Ekkor folytatjuk a módszert egy-egy gyűrűvel mindig kijjebb haladva. A modernebb lakatoknál ez már nem működik a pontosabb gyártástechnológia miatt.

A második módszer, amikor kipróbáljuk az összes lehetséges jelszót, hogy be tudjunk lépni (filmekben látunk ilyet), de emiatt korlátozzák a próbálgatások számát akár adott időre vonatkoztatva, akár végleges kizárás ígérve a rendszerből.

A kulcs egyszerű megjegyezhetősége fontos, mert a feladónak és a címzettnek egyaránt ismernie kell.

A monoalfabetikus kódolásnál, amikor nem Caesar eltolt kódot használunk, hanem a Kámaszutra véletlenszerű keverés módszerét, akkor meg kell jegyeznünk, hogy pontosan melyik betűt, mivel helyettesítettük. Ez jóval nehezebb feladat, mint az ABC ismeretében csak az eltolás mértékét megjegyezni és ezáltal könnyen leírni bármikor a nyílt és kódabécét.

Ha nehéz megjegyezni, megnő a veszélye, hogy az alkalmazója leírja őket és megtalálva elolvashatóvá válnak az üzenetek.

Nézzünk egy módszert a könnyebb megjegyezhetőségre a *kulcsszó* vagy *kulcsmondat* használatát.

Ha például JULIUS CAESAR nevét használjuk kulcsmondatként, akkor eltávolítva az ismétlődő betűket (JULISCAER), és a fennmaradó betűket alkalmazzuk a kódábécé első elemeiként a többi, nem felhasznált betűvel folytatva. Megállapodás kérdése a feladó és a címzett között, hogy a kulcsszó, kulcsmondat betűi után az ABC maradék betűit az elejéről, vagy pedig az utolsó kulcsként alkalmazott betűtől kezdve használjuk fel a kódabécében.

Az utóbbit választva a következő kódábécét kapjuk:

```
Nyílt ábécé:      abcdefghijklmnopqrstuvwxyz
Kódábécé          JULISCAERTVWXYZBDFGHKMNOPQ
```

A címzettnek és a feladónak csak a *kulcsszó* vagy *kulcsmondat* megjegyzésével kell terhelni a az agyát, ennek és az ABC ismeretében mind a nyílt, mind a kódábécé bármikor előállítható, a kódolás, dekódolás elvégezhető és a leírt kódabécé megsemmisíthető.

A *kulcsszó* vagy *kulcsmondat* alkalmazása ugyan lényegesen csökkenti a $403*10^{24}$ lehetőség számát, mégis olyan mértékű marad, hogy továbbra is megfelelő biztonságot nyújt a kódolt üzenetnek.

Középkor

A középkor nagy részében elterjedten a monoalfabetikus kódolást használták, amikor egy betűt egy másik betűvel, vagy szimbólummal helyettesítjük a titkosított üzenetünkben.

Egy pár példát rögtön az elején meg is nézhetünk.

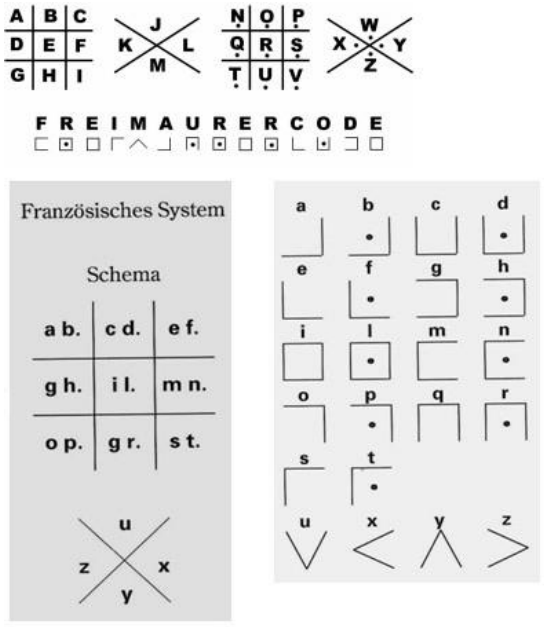
Nagy Károly kódolása

Láthatjuk, hogy Nagy Károly milyen összetett szimbólumokat használt a betűk helyettesítésére. Az is feltűnhet, hogy a szimbólumok használata időigényesebb egy levél megírásánál és jobban kell figyelnünk arra, hogy szépen rajzoljuk le őket, nehogy félreérthető legyen és egy másik szimbólumnak gondolják. (12. ábra).

12. ábra Nagy Károly kódábécéje [3]

Freimauer-féle kódolás

Érdekes a Freimauer-féle kódolás, ahogy 2 db 3x3-as és 2 db 2x2-es alakzat pozíciójába írják be az ABC betűit és helyettük a keretezővonalat rajzolják le (13. ábra). Az ábrán alsó részében az is látható, hogy a francia alkalmazás hasonló, de máshogy sorolja be a betűket, ezért az azonos kerettel jelölt szimbólum más-más betűt jelöl a két nyelvben.



13. ábra A Freimauer-féle kódolás [3]

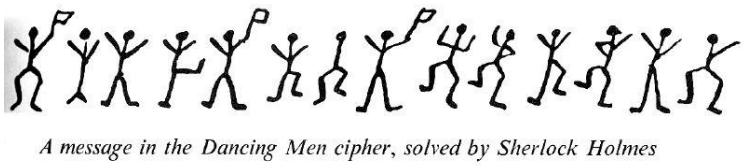
IV. Rudolf osztrák herceg kódolása

IV. Rudolf osztrák herceg által használt szimbólumok esetében feltűnhet, hogy nem feltétlenül egy betű-egy szimbólum megfeleltetés van, de erről majd később (14. ábra).

14. ábra IV. Rudolf osztrák herceg (1339-1365) által használt monoalfabetikus kód [3]

Sherlock Holmes visszatér

Sir Arthur Conan Doyle: Sherlock Holmes visszatér c. regényében is találunk példát a betűk szimbólummal történő helyettesítésére (15. ábra).



15. ábra Titkosított üzenet a Sherlock Holmes visszatér c. könyvben

A figurák végtagjainak helyzete, illetve a zászló megléte, vagy hiánya más-más betűt kódol alakonként. Az ember egy ilyen üzenetet látva visszariad és feltörhetetlennek gondolja, de a következő rész pont azt világítja meg, hogy mindegy, hogy karaktereket, vagy szimbólumokat látunk, ugyanazzal módszerrel lesznek megtörhetők ezek a titkosított szövegek.

Kriptoanalízis megjelenése

Mohamed próféta által elindított változások (isz. VI. sz.) nem csak vallási, hanem annak következtében társadalmi változásokat is hoztak, mely együtt járt a hódítások megindulásával az Arab-félszigetről lefedve Afrika északi részét és a Ibériai-félsziget déli felét. A hóditások után egy viszonylagos béke köszönt be, mely magával hozta a tudomány felemelkedését ezeken a területeken.

Matematikában, csillagászatban, orvostudományban olyan előre lépések történtek, melyek később kihatottak az akkor tudomány terén elmaradottnak számító Európa történelmére is, sőt Európából mentek orvosok, tudósok tanulni az arab tudósokhoz.

A Korán olvasása minden igazhitű számára kötelező az iszlám vallás szerint és nem is tudjuk elképzelni, hogy ennek milyen hatása lesz a kriptográfia területére.

Az arab teológusok a Korán tanulmányozásánál nem csak a szavakat, hanem az egyes betűket is elemezték, s ennek során azt tapasztalták, hogy bizonyos betűk gyakoribbak a többieknél. Eme felfedezés révén vált elérhetővé a behelyettesítéssel titkosított szöveg belátható időn belüli feltörése, melynél a betűk gyakoriságának ismerete elengedhetetlenül szükséges.

A IX. században már az arabok használták is a módszert, míg Európában erre még kb. 500 évet várni kellett. Ott ugyanis továbbra is tartotta magát az a hit, hogy a $403*10^{24}$ lehetőséget kínáló betűcserés módszer feltörhetetlen. Miért érdekes ez? Gondoljuk végig, ezt pont a Kereszteshábórúk időszakára esik. Tehát az arabok képesek elolvasni azt a titkosított üzenetet, amit az európaiak egy biztosnak gondolt módszerrel titkosítanak és meg vannak arról győződve, hogy hiába fogják el a futárt, lövik le a postagalambot, az ellenség nem képes annak elolvasására. *De! Csak nem mondták el!* Nézzük is meg, hogyan lehetséges az, amit mi lehetetlennek tartottunk.

Az arab tudósok már nemcsak a kódábécék alkalmazására, hanem a *megfejtésükre* is képesek voltak.

Ők találták fel a *kriptoanalízist,* a kódszövegek kulcs ismerete nélküli megfejtésének tudományát.

Abu Al-Kindi (801-873)
A kriptoanalízis atyja, matematikus, filozófus, orvos.
Titkos nyelv c. könyv szerzője
*Először használta a 0 számjegyet, az indiai számjegyek bevezetésének meghatározó szereplője volt. Munkáit a helyi könyvtárban őrizték a 13. századi mongol inváziói*g, *minek következtében elpusztultak.*

A monoalfabetikus kódolás törése

Al Kindi által kidolgozott módszer esetén a monoalfabetikus (egy betűt egy másik betűvel, vagy szimbólummal helyetettesítünk) titkosítás megfejtéshez ismernünk kell, milyen nyelven íródott a kódolt üzenet. Ezt a futár élve való elfogásával (1907-ben elfogadott hágai konvencióról még nem történt komolyabb egyeztetés ebben az időben) és kikérdezésével hamar kideríthető.

A nyelv ismeretében veszünk egy azon nyelven írt nyílt szöveget, amely elég hosszú (kb. 1 oldal), hogy megállapítsuk, melyik betű hányszor fordul elő benne.

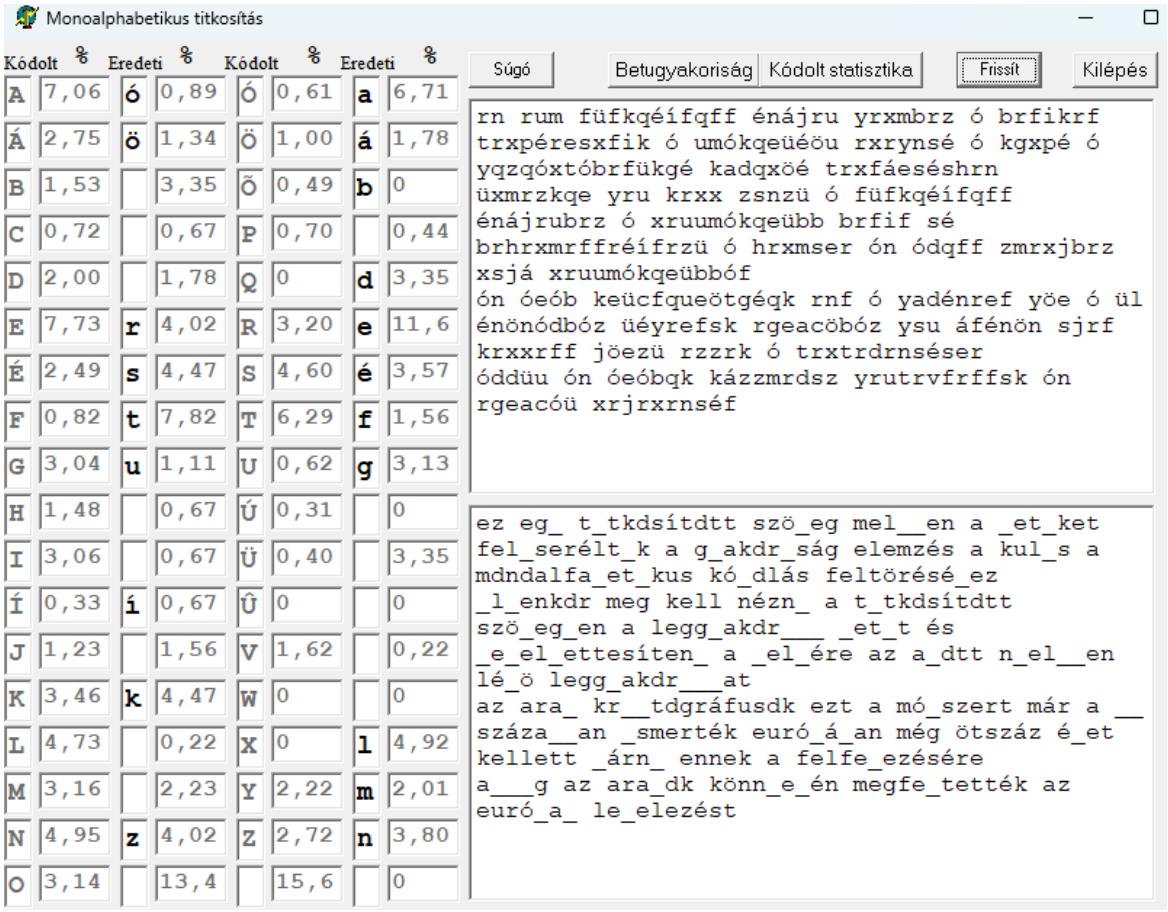
Ezután vesszük a kódszöveget s annak betűit/szimbólumait is megszámláljuk.

Megkeressük a legsűrűbben előfordulót, s azt behelyettesítjük a nyílt szöveg leggyakoribb betűjével, a sorban következő leggyakoribbat a „második”-kal, az az utánit a „harmadik”-kal, és így tovább, míg be nem helyettesítjük a megfejteni kívánt szöveg minden szimbólumát.

Az eljárás jellemzően 3-5 karakterig könnyen megy, a kiugró gyakoriságú betűket (pl. e, a ,t a magyar nyelvben) könnyen beazonosítjuk. Ezután viszont már szótöredékeket kapunk, melyek segítségével a többi szimbólum jelentése is megállapítható.

Bármelyik természetes nyelvben a szóköz gyakorisága a legnagyobb (közel 2x akkora, mint a leggyakoribb betűé), ezért a kriptogram/kódolt szöveg „betűinek” gyakoriság-elemzésével könnyen meg lehet állapítani, melyik „betű” helyettesíti a szóközt (amennyiben van benne).

Látjuk már mennyire egyszerű és zseniális az ötlet?



16. ábra Gyakoriságelemzés a gyakorlatban

A feltörés közbeni állapotot láthatjuk az erre célra kifejlesztett programban (16. ábra).

A páratlan oszlopok tartalmazzák a magyar nyelvre jellemző gyakoriságértékeket a Toldi szövege alapján kiszámítva.

A páros oszlopok pedig a titkosított szöveg gyakorisági értékeit.

A jobb oldal felső részében a titkosított szöveg látható, alatta pedig a már részben visszafejtett szöveg.

A bal oldalon látható a gyakoriságértékek melle, hogy melyik karaktert a titkosított üzenetben, melyik karakterrel helyettesítettük a feltört szövegben.

Európai kriptoanalízis

Európa a korábban említett okoknál és egyéb történelmi körülmény miatt lemaradt a kriptoanalízis terén, megnyugtatta őket a $403*10^{24}$ lehetőséggel kecsegtető monoalfabetikus kódolás biztonsága. Mivel nem jöttek rá, hogy a betűk előfordulásának gyakoriságában van különbség (kevesen is tudtak olvasni) a betűgyakoriságelemzés módszerét nem dolgozták ki.

A IX. és a XIII. század idején Európában csupán a kolostorokban folyt a titkosírás tanulmányozása, ahol a szerzetesek a Biblia rejtett jelentéseit keresték.

Ótestamentum több helyen utal is a titkosírásra. Tartalmaz egyebek közt olyan szövegrészleteket, amelyeket *atbassal,* hagyományos héber behelyettesítéses kóddal írtak. Ennek lényege, hogy az ABC első betűjét az utolsóval helyettesítik, a másodikat az utolsó előttivel, stb. Ebből is ered a neve:

```
A  ->   T,        B    ->   S

"álef", „táv",    "bét", „sin"
```

Az írásjelek elhagyása is működő módszer lehet bizonyos esetekben az igazi jelentés elfedésére.

Amikor a II. András (1205-1235) felesége, Gertrúd királynő ellen szervezkedők kikérték János, esztergomi érsek véleményét, ő az alábbi választ küldte el:

"A királynét megölni nem kell félnetek jó lesz. Ha mindenki egyetért én nem ellenzem."

A hiányzó vesszők beillesztése különböző értelmezést adhat.
Az egyik változat, amikor beillesztjük a vesszőket:

„A királynét megölni nem kell, félnetek jó lesz; ha mindenki egyetért én nem, ellenzem."

Ezzel bizonyítható lenne, hogy az érsek ellenezte a szervezkedést.

A másik esetben a királyné és idegen származású rokonai ellen összeesküvők támogatásaként is felfogható:

„A királynét megölni nem kell félnetek, jó lesz; ha mindenki egyetért, én nem ellenzem."

Mindenesetre a nem egyértelmű nyelvi forma használatával az érsek elkerülhette az esetleges felelősségre vonást, függetlenül az 1213-ban elkövetett pilisi erdei merénylet kimenetének sikerességétől.

A középkor vége, az újkor kezdete

A XV. századi Európában a kriptográfia egy kisebb kör számára már virágzó iparággá fejlődött. Ebben az időben az itáliai független városállamok mindegyike igyekezett előnybe kerülni a többivel szemben. Állandóan szövetséget kötöttek és bontottak fel, konspiráltak, tehát minden körülmény adott volt ahhoz, hogy nagyon fontos legyen az üzenetek titkosítása, hogy elfogásuk esetén ne derüljön fény az eredeti tartalmukra. Emiatt minden államnak megvolt a maga kódja, és minden nagykövetnek volt kriptográfus titkára.

Giovanni Soro volt az első nagy európai kriptográfus, akit 1506-ban neveztek ki Velencében titoknoknak. A baráti államok hozzá, küldték elemzésre az elfogott üzeneteket. Még a Vatikán is hozzá folyamodott a látszólag megfejthetetlen kódokkal. Soro ugyanis rájött arra, amire az Arab kriptográfusok már. IX. században és sikerrel használta a betűgyakoriságelemzést a különböző monoalfabetikus kódolások feltörése során.

Soro-t 1506-ban kérték fel, amikor I. Miksa német-római császár csapatai már a Velencei Köztársaságot fenyegették.

Soro megfejtette az ellenséges Colonna főparancsnok levelét, melyben 20.000 dukátot követelt a császártól, vagy azt, hogy személyesen jelenjen meg a csatatéren. Ebből kiderült, hogy a főparancsnoknak már nincs elég erőforrása a harc folytatására.

Soro 1542-ben két munkatársával a Dózse-palotában dolgozott egy irodában, amit addig nem hagyhattak el, amíg nem dekódolták a rájuk bízott iratot.

Újkor

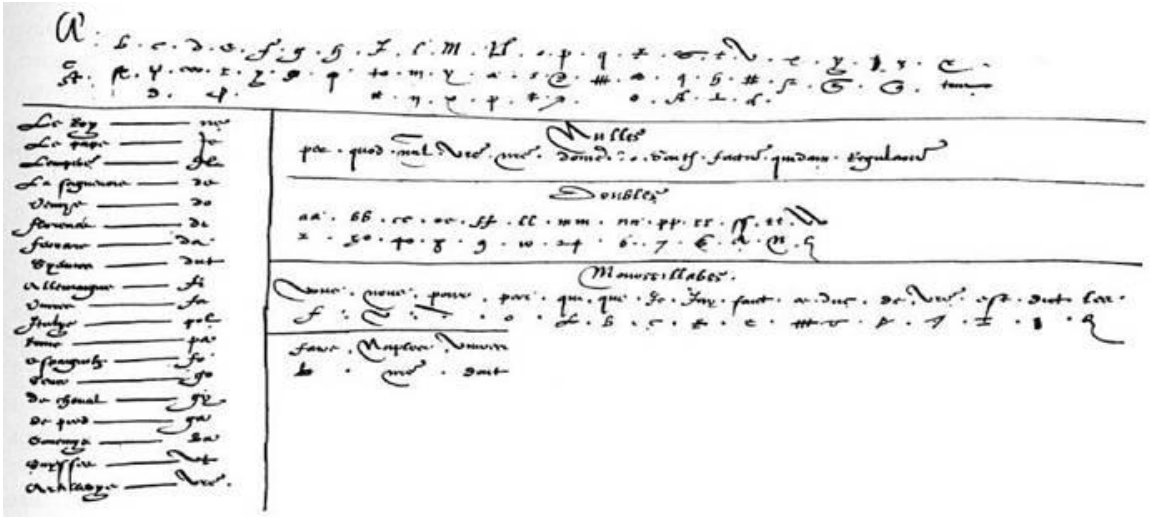
A XVI. század vége felé Francois Viete a francia rejtjelfejtő elsősorban a spanyol kódok feltörésében jeleskedett. Ő is rájött a gyakoriságelemzés módszerére és sikerrel alkalmazta a velük akkor szemben álló spanyolok kódolt üzeneteinek megfejtése során. A spanyolok rájöttek, hogy Viette el tudja olvasni a kódolt leveleiket, de arra, hogy hogyan, nem. II. Fülöp spanyol király végül a Vatikánhoz fordult segítségért utalva arra, hogy Viete képességére egy magyarázat lehet, hogy lepaktált az ördöggel. Ezzel szorgalmazta, hogy Viete-et állítsák a bíborosok tanácsa elé és tárgyalják meg az ügyét (feltételezve az elmarasztaló és Viete életét kioltó ítélet meghozatalában). A pápa elutasította a kérést, mert ekkorra már a Vatikán titoknokai számára is ismert volt a betűgyakoriságelemzés és ők is gond nélkül olvastál a spanyol levelezést, csak erről az érintetteket elfelejtették értesíteni.

Nomenklátorok

Azokban az országokban, amelyekben már felismerték a monoalfabetikus kódolás gyengeségeit, igyekeztek kidolgozni az ellenség rejtjelfejtői képességének jobban ellenálló titkosírásokat, hasonlóan ahhoz, ahogy most tesszük a kvantumszámítógépek megjelenésének árnyékában.

Bevezették a *nullitásokat,* amelyek nem jelöltek semmit, de a gyakoriságértékekre kihatottak. Alkalmazták még rossz helyesírás módszerét, így lassítva a megfejtést Bevezették olyan szimbólumok használatát, melyek nem betűt, hanem önálló szavakat (kódszavak) helyettesítettek.

Ezek a változások együttesen a *nomenklátorok* megjelenéséhez vezettek (17. ábra).



17. ábra Cosimo de Medici által használt nomenklátor (1554) [3]

A nomenklátor egy kódábécé alapú rejtjelezési rendszer (ezt használják az üzenet nagyobb részének titkosítására), amelyet kódszavak korlátozott hosszúságú listája egészít ki. A kódszavak alkalmazásának ellenére a nomenklátor nem nyújt sokkal nagyobb biztonságot az

egyszerű kódábécénél, mivel az üzenet legnagyobb része gyakorisági elemzéssel megfejthető, a fennmaradó kódszavak jelentése pedig a szövegkörnyezetből kikövetkeztethető.

Egy érdekes történelmi esemény kapcsolódik a nomenklátorokhoz, Mária Skócia királynéja, Franciaország királynéja, angol trónkövetelő kapcsán.

Mária fenyegetést jelentett Erzsébetre, mivel az angol katolikusok Máriát tekintették Anglia igazi királynőjének. Nagyanyja, Tudor Margit, VIII. Henrik nővére révén Mária valóban jogot formálhatott a trónra, de Henrik utolsó életben maradt leszármazottjának, Erzsébetnek az igénye elsődlegesnek látszott. A katolikusok mindazonáltal azt állították, hogy Erzsébet nem jogosult a trónra, mivel anyja VIII. Henrik második felesége, Boleyn Anna volt, akit a király az után vett feleségül, miután Aragóniai Katalintól a pápai tilalom ellenére elvált.

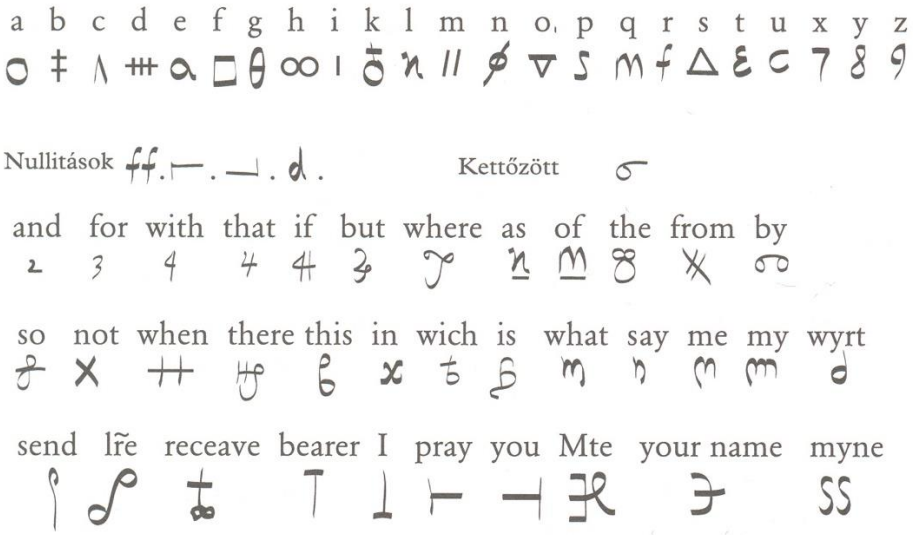
Érdekes történelmi tény, hogy a franciák szerettek volna Hódító Vilmos után újra angliai területeket meghódítani és a Skótokra számíthattak ezen céljuk elérésében. Egyesítette őket a közös vallás és az angolok által velük szemben elkövetett bűnök sorozata. Csak a jelre vártak, mikor törjenek be Angliába mind a tenger, mind a szárazföl felől. A jel Erzsébet királynő eltávolítása a trónról.

Mária 1586-tól a staffordshire-i Chartley Hallban volt őrizet alarr. Kiszabadítását Anthony Babington tervezte meg, aki titkosított üzenetek révén tartotta vele a kapcsolatot. Az egyik levelében Babington jelezte, hogy mivel Erzsébetet V. Pius pápa kiátkozta, ezért véleménye szerint a királynő meggyilkolható, így letaszítva a trónról és helyére Máriát ültetve. ária válaszolt Babingtonnak, említést tett a "fondorlatról", és kérte Babingtont, hogy Erzsébet meggyilkolásakor, vagy még előtte szabadítsák ki.

Az üzenetek titkosítását huszonhárom szimbólum adott betűnek való megfeleltetés alapozta meg (a **j, v** és **w** betűk kimaradtak), továbbá harmincöt további szimbólumból, amelyek szavakat és kifejezéseket helyettesítettek. Négy nullitást is tartlamazott a nomenklátor, valamint egy olyan szimbólumot, amely azt jelezte, hogy a következő betű kettőzött betű (18. ábra).

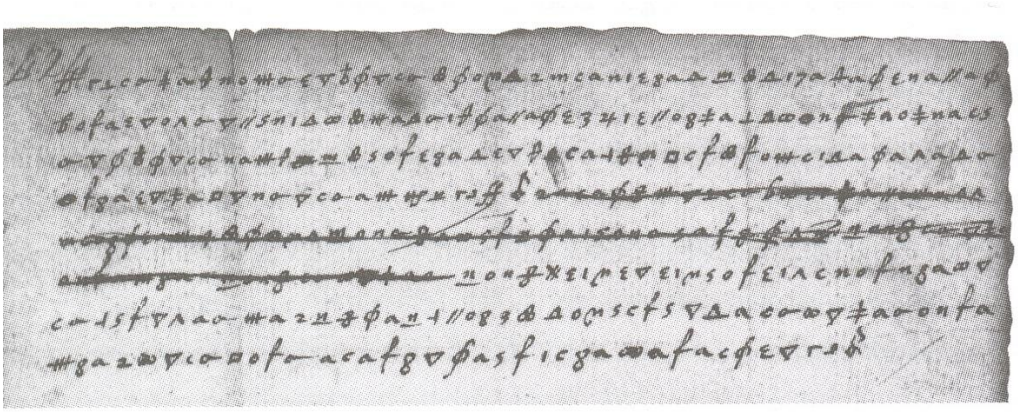
Erzsébet titkára, Walsingham, titoknokként, Thomas Phelippes alkalmazta, aki igyekezett a Gilbert Gifford kettősügynök által szolgáltatott titkosított leveleket feltörni és ezáltal bizonyítékot szolgáltatni a felségárulás tényére. Gifford a Máriát támogatókat korábban meggyőzte arról, hogy el tudja juttatni Máriához a leveleket és a válaszokkal vissza is tér, csak elfelejtette közölni, hogy kerülőúton teszi mindezt, eljuttatva a leveleket Erzsébethez.

Phelippes végül sikerrel járt, megnézve az alkalmazott nomenklátort, az üzenet jelentős része a 23 betűt jelölő szimbólum gyakoriságértékeinek meghatározásával megfejthető, a szavakat jelölő szimbólumok ezután már könnyen szóhoz rendelhetők a szövegkörnyezet ismeretében.



18. ábra Mária skót királynő nomentkátora [1]

Walsingham kezében Mária válaszát megfejtve ott volt a bizonyíték a felségárulásra, de szerette volna elfogni az összesküvőket is, így Phelippest megkérte, hogy hamisítson egy utóiratot Mária levelére, megkérve Babingtont, ismertesse azokat, akiknek Mária a kiszabadulása után hálás lehet és méltó jutalomban részesíthető őket (19. ábra).



19. ábra A hamisított utóiratot tartalmazó levél [1]

A történet lezárásaként, a sikeresen megtört titkosított levelezés alapján, Máriát elítélték és 1587. február 8-án fogsága színhelyén, a kastély egyik termében, háromszáz meghívott vendég jelenlétében lefejezték.

Kódszavas kódolás

A kódszavas titkosításnál használt módszer, de van két hátránya a monoalfabetikus behelyettesítéshez képest.

Az első, hogy a feladó és a címzett ki kell dolgozzon egy szóbehelyettesítő kódot (pl. Atreides harci nyelv), szótárt, amelyben a nyílt szövegben adott körülmények között használható szavakat, illetve azok kódszavait jelenítik meg.

A második hátrány, ha a kódkönyvet megszerzi az ellenség, minden korábbi kódolt üzenetváltás megfejthetővé válik. A feladó és a címzett kénytelen egy új kódkönyvet összeállítani és mindenkinek eljuttatni, aki az üzenetváltásokban részt vesz.

Biliterális, polyliterális behelyettesítés

A nyílt ábécé egy-egy betűjét több karakter helyettesíti a kódolásnál. Kétbetűs helyettesítéskor *biliterális,* több betűs helyettesítéskor *polyliterális* rendszert kapunk.

Már az ókorban *Polybios* is javasolta, hogy az ábécé betűit egy mátrixba rendezzék és a nyílt szöveg karakterei helyett a mátrixnak azt a sorát és oszlopát adják meg egy-egy betűvel/számmal jelölve, amelyben a nyílt szöveg adott betűje szerepel (20. ábra).

	A	**B**	**C**	**D**	**E**	**F**
A	a	b	c	d	e	é
B	f	g	h	i	Í	j
C	k	l	m	n	o	ó
D	ö	ő	p	r	s	t
E	u	ú	ü	ű	v	y
F	z	.	,	:	?	

20. ábra Polybios féle betűmátrix [1]

Ha a címzett és a feladó megegyezzik abban, hogy minden kétbetűs helyettesítésnél az első betű a mátrix sorát, a második pedig a mátrix oszlopát jelöli, akkor pl. a nyílt szöveg **p** betűjét a DC betűpár helyettesíti, a szóközt pedig az FF.

Ez a behelyettesítés is monoalfabetikus kódolásnak számít, csak a nyílt szöveg minden egyes betűje helyett két-két betű áll a kódolt szövegben. A kriptogram betűstatisztikája azonban azonnal kimutatja, hogy összesen csak 6 betű fordul elő benne és az is nyilvánvaló, hogy betűpárok fognak ismétlődni, méghozzá nem azonos, hanem a nyílt szöveg karaktereinek megfelelő gyakorisággal. Ha ezt felismerjük, akkor a kriptogram már vissza is vezethető az egyszerű, monoalfabetikus helyettesítésre.

Tehát az üzenet elfogójának csak arra kell rájönnie, hogy minden üzenet páros számú karakterből áll, az egymásutáni betűpárokra gyakoriságelemzést végezve megfejthető az eredeti szöveg.

Tritheimius abbé (1462-1516) dolgozta ki a három szimbólumból álló kódolást. Az 1,2 és 3 számjegyekből összesen $3^3 = 27$ permutáció képezhető (111, 112, … …333) és ezekkel rendre helyettesíthető a 26 betűs angol ABC egy kivétellel

Sir Francis *Bacon* a nyílt ábécé minden egyes betűjéhez egy ötkarakteres kódot rendelt hozzá, melynél mindössze két szimbólumot használt.

A három, illetve öt karakteres behelyettesítés hasonló módon megtörhető a gyakoriságelemzés módszerével, mint a két karakteres, így ezek plusz védelmet nem nyújtanak.

A geometriai titkosítás

A monoalfabetikus kódolás során a betűk megtartották eredeti pozíciójukat a más betűre, vagy szimbólumra való kicserélésük során. Ez a tény segítette a kriptográfusokat a visszafejés során, hiszen szórészletek ismeretében már könnyebben azonosíthatók a hiányzó betűket jelölő szimbólumok.

A transzpozíció, amikor az eredeti üzenet betűi elveszítik eredeti pozíciójukat, egy jól kezelhető módja az un. *geometriai titkosítás*.

A nyílt szöveg betűit egy (geometriai) rácsszerkezetbe írják be, és más sorrendben olvassák ki ugyanabból a rácsszerkezetből (lásd. pl. füles újság feladatai). A beírási és kiolvasási módszerek igen változatosak (átlósan, függőlegesen, csigavonalban, stb.).

A geometriai titkosítás esetében a betűgyakoriságok pontosan megegyeznek a nyílt szöveg betűgyakoriságaival, mivel csak pozícióváltozás történik, illetve elvész a szomszédosság.

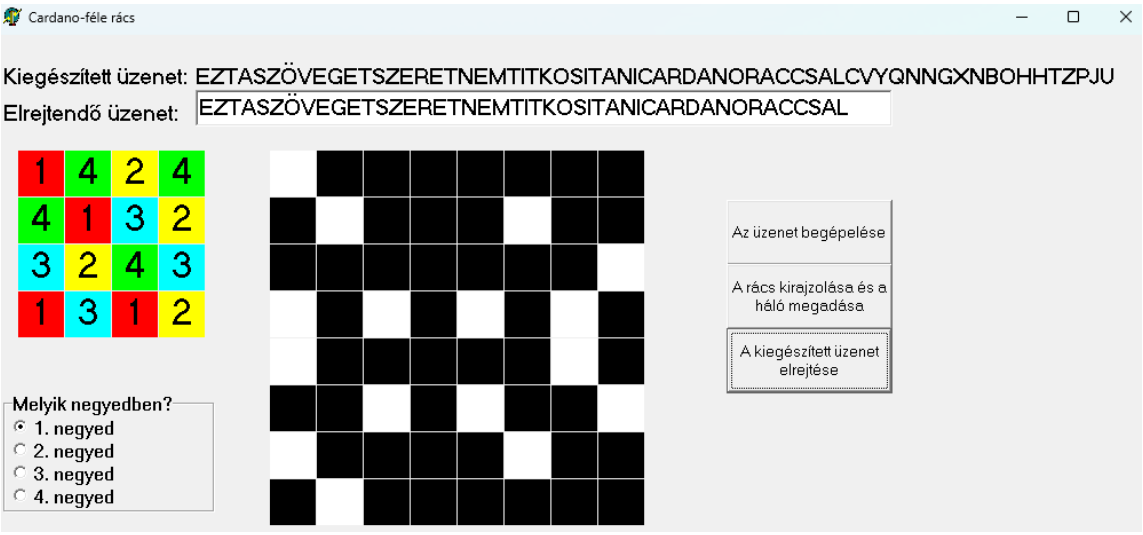
Az így előállított kriptogram feltöréséhez „gyanús szavakat" keressük úgy, hogy feltételezzük, hogy azok betűi egymástól azonos távolságra kerültek az összekeveréskor. Amennyiben a transzpozíciót egymás után többször végzik el, az ezt is megváltoztatja.

Kicsit megint előre szaladva, a modern kriptorendszerekben egy un. permutációs mátrix a keverés kulcsa, melynek működése hasonlít a geometriai titkosításhoz.

Forgatható, Cardano-rács

Egy négyzetes mátrixba beírt szöveg egy forgatható ráccsal is összekeverhető (lásd Jules Verne: Sándor Mátyás c. könyve). Ehhez egy sablont használnak, mely a rácsra ráhelyezve eltakarja annak jelentős részét, de bizonyos helyek lyukasak rajta, ezért az alatta lévő hely a mátrixban látható. Ezekbe a pozíciókba írjuk be soronként haladva, egyesével az üzenet betűit, amint elfogytak a helyek, a sablon a címzettel korábban egyeztetett irányban 90 fokkal elforgatjuk, így újabb helyek lesznek láthatók és használhatók a mátrixban betűk beírására. A forgatást összesen 3x tudjuk elvégezni, utána az eredeti állapotba jutunk. A sablont elvéve a látható lesz az üzenetünk minden betűje, de össze lesznek keverve a sablon alapján, elvész a szomszédossága a szövegnek.

Egy páros elemszámú négyzetes mátrixban mindig kijelölhető az elemszám egynegyedének megfelelő cella úgy, hogy a mátrix transzponálásakor egyik ilyen megjelölt elem se kerüljön egy másik helyére [4].



21. ábra Cardano-rács forgatható sablonjának kialakítása

A kifejlesztett programban először beírjuk a titkosítandó üzenetet. A Cardano-rács csak páros számú oldalakkal rendelkezhet a forgathatóság miatt. Az üzenetet véletlenszerű karkaterekkel egészítjük ki addig, amíg a szükséges méretű mátrix (ez a szöveg hosszától függ, nem fix!) minden egyes pozíciója ki nem lesz töltve. Ha maradnának üres helyek, abból könnyen ki lehetne találni a sablon szerkezetét.

A bal oldalon, ahol különböző hátterű színnel találunk számokat, a sablon kilyukasztásának helyeit adjuk meg a sablon egy negyedére lebontva. Az 1-esek a mátrix bal felső negyedében (első negyed) jelzik a lyukak helyét (21. ábra). a 2-es számok a jobb felső negyedben (második negyed), de már 90 fokkal elforgatva. A 3-asok a jobb alsó negyedben 180 fokkal elforgatva, a 4-esek a bal alsó negyedet jelölik, ahova 270 fokkal elforgatottan kerülnek a lyukak. Ezzel a módszerrel biztosítjuk a forgatható sablont úgy, hogy azonos helyet a mátrixban ne fedjen fel többször.

22. ábra Cardano-rács forgatható sablonjának kialakítása

A teljes szöveg sablonba történő beírása után láthatjuk, hogy is néz ki a mátrixban a szövegünk (22. ábra). A visszafejtéshez a címzettnek egy ugyanezen a helyeken kilyukaszott

sablonnal kell rendelkeznie, ráhelyezi a mártixra és szépen soroknént olvassa össze a lyukakban látható betűket, amikor a végére ért, elfordítja a sablont és olvassa tovább az üzenet folytatását.

A töréshez meg kell sejtenünk a feladó és a címzett ismertében, hogy milyen szavak rejtőzhetnek a mátrixban és ezeknek a betűit megkeresve próbálhatunk a sablon kilyukasztási helyeire támpontot találni.

Több kódabécé használata

Leon Battista Alberti a XV. sz. hatvanas éveiben két, vagy három kódábécét javasolt az üzenet titkosításához, azokat minden egyes betű kódolásánál felváltva használva a gyakoriságelemzést ellehetetleníthetjük.

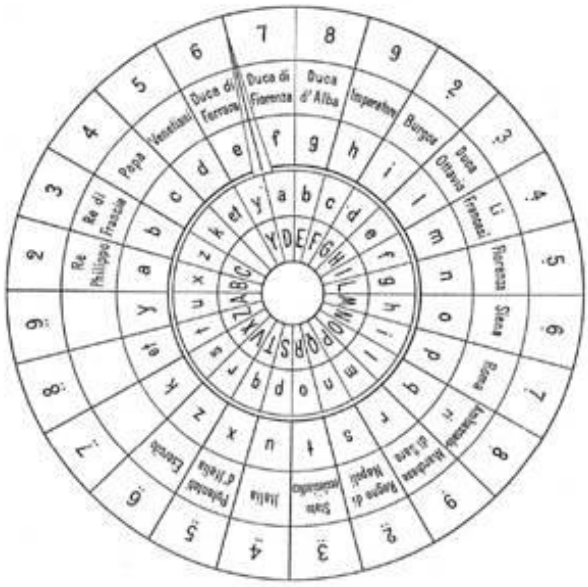
```
Nyílt abc:   abcdefghijklmnopqrstuvwxyz
1. kódabc:   FZBVKIXAYMEPLSDHJORGNQCUTW
2. kódabc:   GOXBFWTHQILAPZJDESVYCRKUHN
```

A **HELLO** szó rejtjelezésekor a páratlan betűket az első, a párosakat a második kódábécével adjuk meg.

A teljes kódszöveg így: **AFPAD**

Alberti által javasolt módszer egyik előnye, hogy az eredeti üzenetben lévő azonos betűk nem feltétlenül lesznek azonosak a kódszövegben.
További hozadékként a kódszöveg két „A” betűje más-más betűt jelöl az eredeti üzenetben.

A több kódábécé használatát a Vatikánban is kidolgozták és a kódoláshoz, visszafejtéshez egy 4 tárcsás korongot is készítettek. A korongok egymáshoz képes elforgathatók, így könnyen módosítható akár minden üzenetváltásnál, hogy melyik betűnek mi felel meg az adott kódábécé esetében (23. ábra).



23. ábra A Vatikánban használt 4 tárcsás rejtjelző korong [3]

Vigenére kódolás

A Blaise de Vigenére által megalkotott rendszer, melyet sokáig feltörhetetlen kódként jegyeztek, 26 különféle kódábécét használ a nyílt szöveg titkosításához. 1586-ban Vigenére az Értekezés a titkosírásról c. dolgozatában foglalta össze a kidolgozott rendszert.

A titkosítás első lépése a Vigenére-tábla elkészítése, melyben a nyílt ábécét 26 kódábécé követ, melyek mindegyikét egy-egy hellyel eltolják az előzőhöz képest. Így az első kódábécé a nyíltábécé egybetűs eltolásával létrehozott Caesar-kód, a második kétbetűs eltolással, stb (24. ábra).

```
nyílt ABC:      abcdefghijklmnopqrstuvwxyz
   1            BCDEFGHIJKLMNOPQRSTUVWXYZA
   2            CDEFGHIJKLMNOPQRSTUVWXYZAB
   3            DEFGHIJKLMNOPQRSTUVWXYZAAC
   4            EFGHIJKLMNOPQRSTUVWXYZABCD.
   .
   24           YZABCDEFGHIJKLMNOPQRSTUVWX
   25           ZABCDEFGHIJKLMNOPQRSTUVWXY
   26           ABCDEFGHIJKLMNOPQRSTUVWXYZ
```

24. ábra Vigenére tábla [1]

A Vigenére kódolás alkalmazásához a címzettnek és a feladónak meg kell állapodnia a sorok váltogatásának sorrendjében.

Ez pl. egy kulcsszó alkalmazásával lehetséges.

Az első teendő az, hogy a kulcsszót a nyílt szöveg fölé írjuk, és annyiszor ismételjük, hogy az üzenet minden egyes betűje hozzárendelhető legyen a kulcsszó valamelyik betűjéhez, ezzel jelölve a kódolásához felhasznált sort:

```
Kulcsszó          W H I T E W H I T E W H I T
Nyílt szöveg      s e n d m o r e t r o o p s
```

A kulcsszó megadja, hogy a Vigenére tábla mely sorait fogjuk használni a kódoláshoz és a visszafejtéshez.

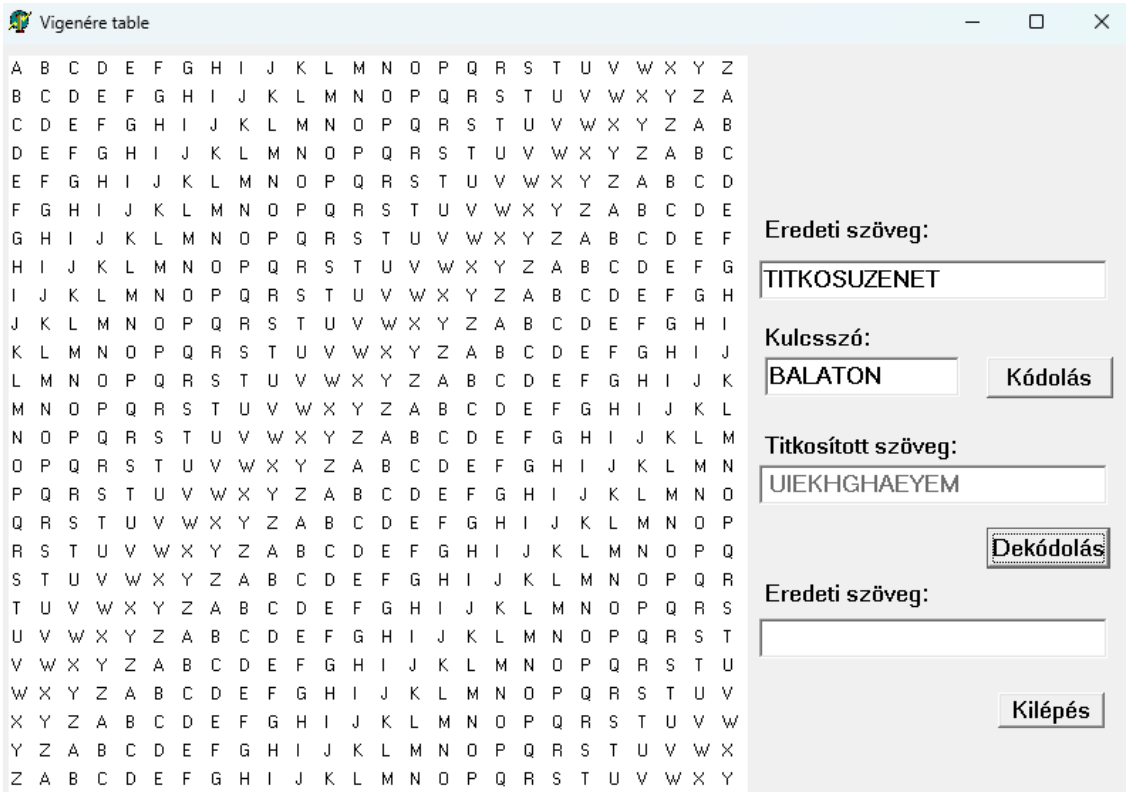
Az első betű, az **s** sifrírozásakor megnézzük, milyen betű van felette. Ez a **W**. Ez meghatározza a Vigenére-tábla W-vel kezdődő sorát, tehát a nyílt szöveg **s** betűjét e szerint a kódábécé szerint helyettesítjük be. Ezután megnézzük, hogy az s-el jelzett oszlop hol keresztezi a W betűvel kezdődő sort: az **O**-nál. Ez lesz a kódszöveg első betűje. A H-val kezdődő sorral kódoljuk a nyílt szöveg második betűjét az **e**-t. Megnézzük, hogy az e-nek mi felel meg ebből a sorból: az **L**. Ez lesz a kódszöveg második betűje, és így tovább (25. ábra).

```
Kulcsszó:         W H I T E W H I T E W H I T
Nyílt szöveg:     s e n d m o r e t r o o p s
Kódszöveg:        O L V W Q K Y M M V K V X L
```

```
nyílt ABC:     abcdefghijklmnopqrstuvwxyz
     4         EFGHIJKLMNOPQRSTUVWXYZABCD
     .
     7         HIJKLMNOPQRSTUVWXYZABCDEFG
     8         IJKLMNOPQRSTUVWXYZABCDEFGH
     .
     19        TUVWXYZABCDEFGHIJKLMNOPQRS
     .
     22        WXYZABCDEFGHIJKLMNOPQRSTUV
```

25. ábra Vigenére kódolás menete [1]

A Vigenére kódolás szemléltető programba beírjuk a titkosítandó üzenetet és a használni kívánt kulcsszót (26. ábra). A program megjelöli a kulcsszó megfelelő betűjével kezdődő sort és a titkosítandó üzenet megfelelő betűjének oszlopát. A két jelölés, a sor és oszlop találkozásánál lévő betű lesz a titkosított üzenet adott betűje.



26. ábra Vigenére kódolást menete [1]

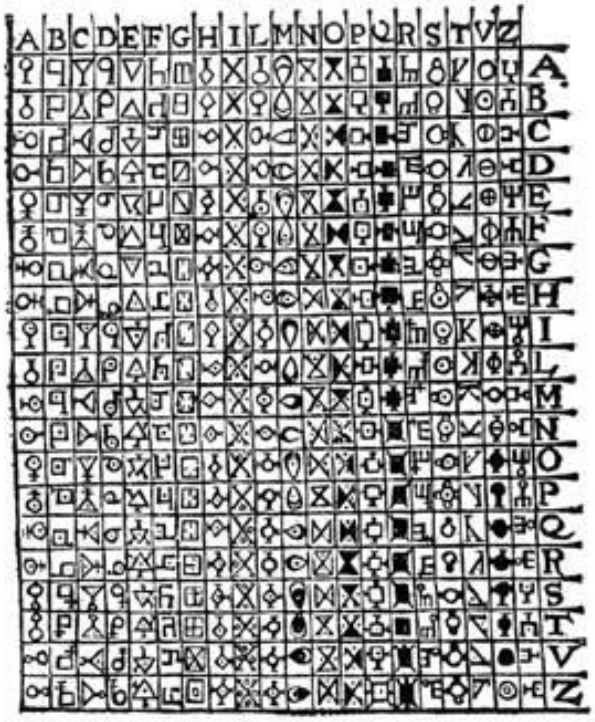
A Vigenére-sifrét nem kezdték el használni és ez a polialfabetikus rendszer kb. kétszáz évre feledésbe merült.

Az ok egyszerű, bonyolult volt a korábbi rendszerekhez képest a használata.

A katonai felhasználás során, ahol a csatában gyors kódolásra és dekódolásra van szükség, ennek az alkalmazása túl sok időt vesz igénybe, ezalatt az idő alatt megváltozhat a harci helyzet, ezáltal a kódolt információ elévül.

A Vigenére kódolást, dekódolást megvalósító Python forráskód a 2. mellékletben található.

Giovanni Battista Porta (1535 – 1615) is kidolgozott egy polialfabetikus rendszert, mely Vigenére-től eltérően a betűk helyett szimbólumokat (több mint 400) használt (27. ábra).



27. ábra Giovanni Battista Porta polialfabetikus rendszere [3]

Első látásra Porta rendszere zavaros, mégis ha oszloponként vizsgáljuk, akkor adott szimbólum elforgatott, pontozott, plusz vonással ellátott változatát találjuk egymás alatt.

A Vigenére-kódolás törése

Hall Brock Thwaites 1854-ben bejelentette, hogy feltalált egy új titkosítási módszert, amelyet szabadalmaztatási célból jelentetett meg a Journal of the Society of Art című lapban. A találmány valójában a Vigenére-sifre volt, melyre Charles Babbage hívta fel a szerkesztőség figyelmét rámutatva, hogy ez a módszer már nagyon régi és a legtöbb, ilyem témával foglalkozó könyvben megtalálható.

Thwaites egy ma challenge-nek megfeleltethető ötlettel állt elő. Ha ez a kódolás olyan régi, akkor Babbage biztos meg is tudja törni és küldött számára egy kódolt üzenetet várva a megfejtést (28. ábra):

WUB**EFIQ**LZURMVOFEHMYMWTIXCGTMPIFKRZUPMVOIRQMMWOZMPULMBNYVQQQMVM
VJLEYMHFEFNZ**PSDLPPSDLP**EVQM**WCXYM**DAVQE**EFIQ**CAYTQO**WCXYM**WMSEMEFCFWY
EYQ**ETRL**IQYCGMTWCWFBSMYFPLRXTQYEEXMRULUKSGWFPTLRQAERLUVPMVYQYCX
TWFQLMTELSFJPQEHMOZCIWCIWFPZSLMAEZIQVLQMZVPPXAWCSMZMORVGVVQSZ**E**
TRLQZPBJAZVQIYXEWWOICCGDWHQMMVOWSGNTJPPPAYBIYBJUTWRLQKLLLMDPYV
ACDCFQNZPIFPPKSDVPTIDGXMQQVEBMQALKEZMGCVKUZKIZBZLIUAMMVZ

28. ábra Thwaites titkosított üzenete [1]

A Vigenére kódolás esetén, amennyiben adott karakterláncot a kulcsszó azonos részével titkosítunk, azonos titkosított karakterláncot kapunk:

```
Kulcsszó:          KINGKINGKINGKINGKINGKING
Nyílt szöveg:      thesunandthemaninthemoon
Kódszöveg:         DPRYEVNTNBUKWIAOXBUKWWBT
```

Erre alapozott Charles Babbage az üzenet megtörésénél, megkeresve az ismétlődő karakterláncokat (28. ábra vastagbetűs részei).

Az ismétlődő karakterláncok közötti karaktertávolságot megszámolta és megnézte azok osztóit (5. táblázat). Azt találta, hogy öt a legkisebb közös osztója az ismétlődések hosszának mindegyik esetben. Ezt az feltételezi, hogy Thwaites egy öt betűből álló kulcsszót használhatott, amit nem ismerünk.

5. táblázat Behelyettesítő kód (példa)

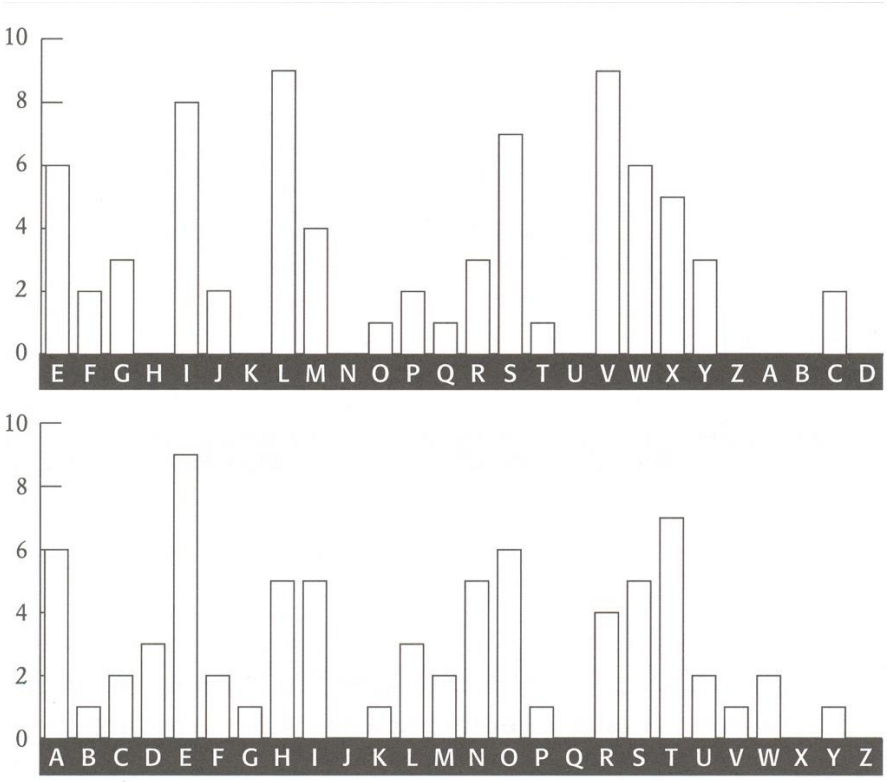
Ismételt betűfüzér	Távolság	A kulcs lehetséges hossza 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 19 20
EFIQ	95	+ +
PSDLP	5	+
WCXYM	20	+ + + + +
ETRL	120	+ + + + + + + + + +

Azt viszont tudhatjuk, hogy egy öt karakteres kulcsszó esetén, az üzenet 1., 6., 11, 16., stb. betűit a kulcsszó első betűjével kezdődő a sorral kódolták a Vigenére táblából. Ugyanígy a 2., 7., 12., 17., stb. betűket a kulcsszó második betűjével kezdődő sorral kódolták.

Tehát nincs más dolgunk, mint az azonos sorral kódolt betűkre gyakoriságelemzést végezni és megnézni, hogyan illeszkednek a gyakoriságértékek a normál szöveg betűgyakoriságához. Ha pl. a legnagobb gyakoriságértékű betű az „E” a nyelvben, az adott gyakoriságtáblázatban megkeressük, melyik a leggyakoribb betű a kódoltak között az adott csoportban és megnézzük, hány karakterrel eltolva jelentkezik ez az érték.

Több ilyen kimagasló gyakoriságérték egymáshoz való helyzete is segíthet az eltolás mértékének megállapításához, ezzel pedig a kulcsszó adott betűjének megfejtéséhez (29. ábra).

Így végighaladva az egyes titkosított betűcsoportokon (a példa szerint minden 5. betű a titkosított szövegből) a teljes kulcsszó megfejtéséhez vezet, mely végül felfedi az eredeti szöveget a Vigenére dekódolás után.



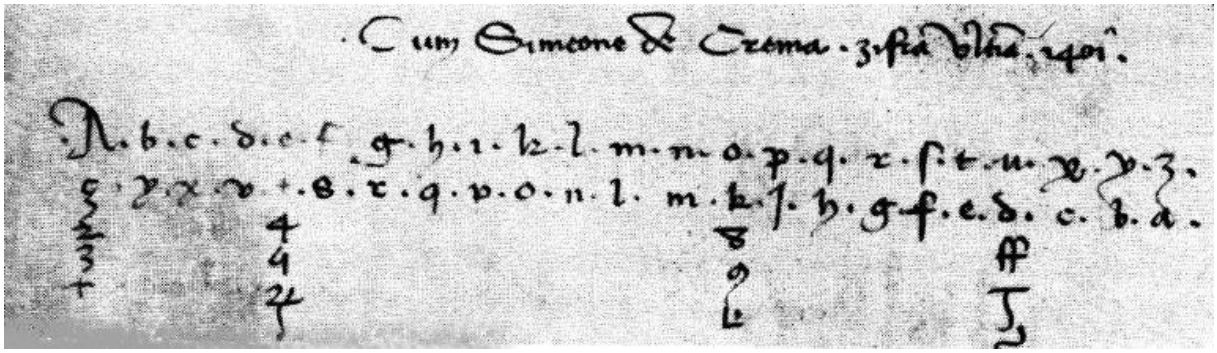
29. ábra Négy betűvel eltolt gyakoriságértékek (felső diagram) [1]

Charles Babbage ugyan sikerrel járt Thwaites titkosított üzenetének megtörését tekintve, viszont nem publikálta. Az íróasztalának fiókjában találták meg a sikeres törés bizonyítékát. 9 évvel később Friedrich Kasiski német tiszt 1863-ban publikálta a saját megoldását, ezért a történelembe már Kasiski törésként vonult be a metódus.

Homofonikus kód

Akik továbbra is gyorsan használható titkosítási módszert szerettek volna használni, de már ismerték a gyakoriságelemzésből fakadó veszélyt, homofonikus behelyettesítésben találhatták meg a számukra megfelelő módszert. Az ötlet a gyakoriságértékek kiegyenlítésén alapszik, akár minden egyes betűnek több sifréje is lehet, melyek száma az adott betű általános gyakoriságához igazodik. Pl. az ***a*** az angolban 8 százalékban fordul elő, akkor 8 különféle szimbólum rendelendő hozzá. Valahányszor megjelenik az ***a*** betű a nyílt szövegben, mindig valamelyik kódjával rejtjelezzük mégpedig a következő módon: az első előforduláskor az első szimbólummal, a második előfordulásnál a másodikkal, stb. helyettesítjük és csak a 9. előfordulásnál használjuk újra az első szimbólumot. Minden betű esetén ezt a módszert követve elérjük, hogy minden szimbólum 1%-ban fordul elő a kódolt üzenetben.

Érdemes megemlíteni, hogy Simeone de Crema 1401-ben már homofonikus kódolást használt, igaz, csak a leggyakrabban előforduló magánhangzókra és nem feltétlenül a gyakoriságértéküknek megfelelő számú szimbólumot használva (30. ábra).



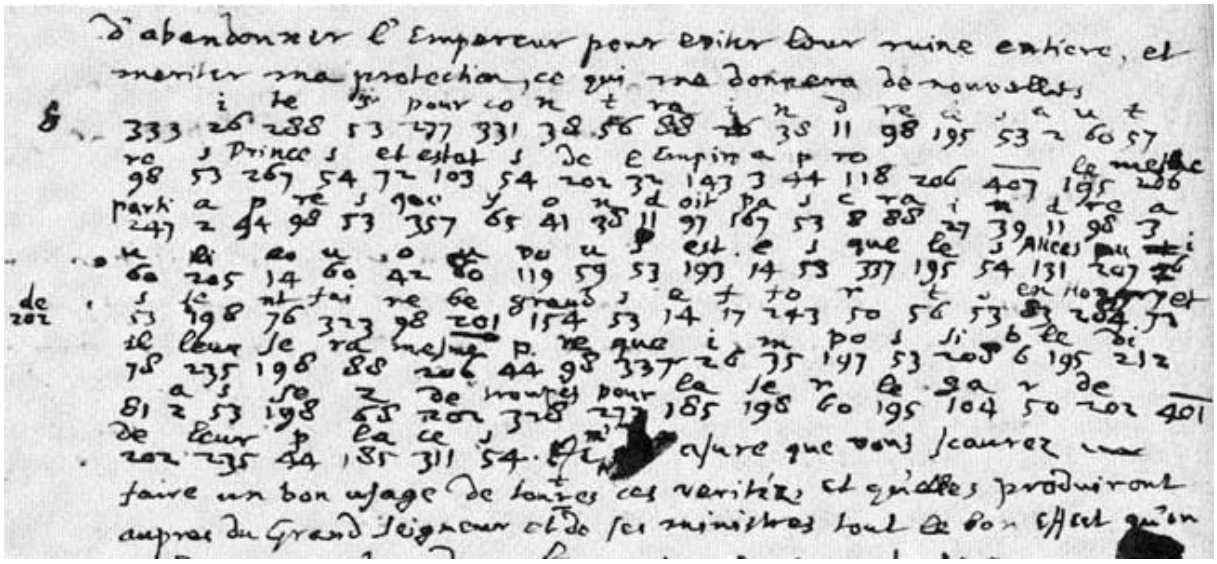
30. ábra Simeone de Crema homofonikus kódja [3]

A Grand Chiffre

A rendszert egy apa-fiú páros, Antoine és Bonaventure Rossignol dolgozta ki és XIII ill. XIV Lajos levelezésének titkosítását végezték el vele.

Az eddigi ismeretek szerint senkinek nem sikerült megtörnie a használata idején.

Apa és fia halála után a nagy kódot nem használták többé, mivel nem avattak be senkit a működésébe. A kódolás 1890-ig feltöretlen maradt.



31. ábra Grand chiffre [1]

Étien Bazeries kezdte el a feltörést egy kezébe került, addig ismeretlen irattól ihletve, mely a nagy kóddal volt sifrírozva.

A kód 587 féle számot tartalmaz (31. ábra). Úgy gondolta, hogy homofonikus kóddal van dolga, de nem jutott eredményre.

Ezután Bazeries azt feltételezte, hogy minden szám egy betűpárt jelöl.

Megkereste a kódszöveg leggyakrabban előforduló számait (22, 42, 124, 125, 341) és megpróbálta a leggyakoribb francia betűpárokkal (es, en, ou, de, nt) behelyettesíteni őket, gyakoriságelemzést alkalmazva betűpárokra.

Mivel így sem jutott eredményre, számokhoz szótagokat próbált társítani, de ez sem tűnt célravezetőnek. Viszont talált egy olyan számcsoportot mely nagyon gyakran szerepelt (124-22-125-46-345). Feltételezte, hogy ez a **les-en-ne-mi-s**, azaz **az ellenség** szót jelöli, mely végül helyesnek bizonyult. Pár szótag ismeretében sikerült szórészleteket felfedni és újabb számhoz újabb megfelelő szótagot rendelni egészen addig, amíg mind a 587 számhoz rendelt egyet.

A grand chiffre szótagkódolást alkalmazott!

Az egyértelműségi pont

A Claude Shannon (1916-2001) által bevezetett fogalom azt jelöli, mi az a minimális hossz egy titkosított szöveg esetében, hogy egyértelműen megfejthető legyen.

A Caesar típusú helyettesítés egyértelműségi pontja 20 betű.

Kódolt üzenet	nyílt üzenet	angol	magyar
xrxtqxm		ejected	oroszok
		elected	akartam
		arabian	eretnek
		anagram	avatnak

A fenti részből látható, hogy Caesar kód esetén több értelmes szöveg is előállhat esetlegesen a brute force alkalmazása esetén. Az egyértelműségi pont arra utal, hogy ilyen esetben mekkora karakterszám esetén lehet kiválasztani közülük a megfelelőt.

Az 1700-as években már minden európai országban volt egy un. „fekete szoba”, a beérkezett üzenetek elemzésének központja.

A legeredményesebb a bécsi Geheime Kabinets-Kanzlei (titkos kabinett-kancellária) volt, ahol olyan hatékony munkarendet dolgoztak ki, hogy a tevékenység nem késleltethette a postát.

A fontosnak, gyanúsnak gondolt leveleket lemásolták, 3 órán belül újra lepecsételték és visszavitték a központi postahivatalba. Az érdemi munka utána kezdődött, a visszafejtés, ha a levél kódolva volt. Ehhez a bécsi fekete szoba kidolgozta a monoalfabetikus kódok minden formájának megfejtését. Azoknak, akik továbbra is titokban akarták tartani a levelezésük tartalmát, rákényszerültek a korábban már felfedezett, de bonyolultabb, cserébe megbízhatóbb Vigenére-tábla alkalmazására.

A Faistos-i korong

A korongot 1908-ban találták egy palota romjai között Kréta déli részén Faistos közelében.

Ie. 1700-ban készítették (Mínoszi kultúra) a 16 cm átmérőjű és 2.1 cm vastag égetett agyagkorongot (32. ábra).

32. ábra Faistos-i korong [1]

Az egyik oldalán 122 a másikon 119 szimbólum látható spirál alakban, 45 különböző szimbólumot tartalmazva.

Az írás a „Lineáris A” nevet kapta.

Máig megfejtetlen, mivel ilyen írással ellátott másik tárgyat még nem találtak, ezáltal az üzenet hossza nem érte el az egyértelműségi pontot. A kutatók emiatt csak sejtik a tartalmát.

Chappe-féle optikai távíró

A Chappe-féle optikai távíró (1793-1835) egy őrház, tetején lévő árbocrúdra egy három-öt méter hosszú, mozgatható kart szereltek fel, amelynek mindkét végéhez két méter hosszú, szintén mozgatható szárny kapcsolódott (33. ábra).

33. ábra Chappe-féle optikai távíró [3]

A szerelvény rácsos szerkezetű, hogy a szél ne tépje le.

A jelzőelemeket az őrházból lehetett mozgatni két fogantyú segítségével.

A fogantyúk a közvetítő rudazat segítségével a középső kart négy különböző állásba, míg a szárnyakat hét-hét különböző állásba tudták állítani. Így összesen 196 különböző jelzést lehetett beállítani. Ez azt jelenti, hogy nem csak betűkhöz, számokhoz lehet egy-egy beállítást rendelni, hanem gyakran használt szavakhoz is, így felgyorsítható a kommunikáció.

Az őrházakat úgy helyezték el, hogy a távírászok távcső segítségével jól lássák a szomszédos torony jelzéseit (kb. 12-25 km domborzati viszonyoktól függően).

A leolvasott jelzést a távírász a fogantyúk segítségével beállította a saját jelzőberendezésén, majd várta a következő jelzést, közben figyelve, hogy a másik szomszédja hozzá igazította-e már a saját őrházának a beállítását, ezzel garantálva, hogy nem szakad meg az üzenetküldési láncban az adott tartalom továbbítása. Ezzel a módszerrel 1 óra alatt akár 1000 km-re is lehetett üzenetet küldeni, mely abban a korban nagyon gyors üzenettovábbításnak számított a váltott lovak útján történő üzenettovábbításhoz képest.

A forradalom idején kiépült a Párizstól Lille-ig húzódó (kb. 200 km), 15 közvetítő állomásból álló szakasz és első üzenetet 1792-ben küldték ezen keresztül.

1852-ben 4800 kilométer hosszan építették ki a hálózatot 556 őrház felépítésével a 29 legnagyobb várost összekötve Párizzsal.

A rádiózás hatása

1894-ben Marconi felfedezte az elektromos áramkörök azon tulajdonságát, hogy áram alá helyezve egy bizonyos távolságra lévő áramkörben elektromos áramot indukál. Ez vezetett a drótnélküli információközléshez, elkezdik használni a rádiót. 1899. december 12-én siekresen küldtek át adatot az Atlanti-óceán egyik oldaláról a másikra. Egészen1924-ig nem értették pontosan, hogyan lehetséges ez, de akkor felfedezték az ionoszférát kb. 60 km magasságban, és azt, hogy visszaveri a rádióhullámokat.

A minden irányban terjedő rádióhullámok segítségével bárhova tudunk adatot küldeni, ami nagyon nagy előny, mivel még a Chappe-féle távirónál is gyorsabb az adatáramlás, viszont cserébe azt sem tudjuk megakadályozni, hogy bárki fogja az adásunkat, akár az ellenség is. Ez nagyban hasonlít ahhoz, ahogy manapság a WI-FI jeleket tudjuk fogni a különböző eszközeinkkel.

1914-1918-ig nem sikerült jó titkosítási megoldást találnia a kriptográfusoknak.

Legújabb kor

A II. Világháború előtti idők

ADFGVX kód

A rádiózás kezdeti időszakának leghíresebb titkosírási megoldása a német ADFGVX kód volt, melyet 1918. március. 5-én kezdtek el használni.

Fontos volt olyan betűket találni, melyek Morse-kódja jelentősen eltér egymástól, ezzel csökkentve a tévedés veszélyét az üzenet kódolásánál és dekódolásánál. Ez a követelmény az A, D, F, G, V, X betűkre volt igaz, innen nyerte a kódolási eljárás a nevét (6. táblázat).

6. táblázat Morze kódja az ADFGVX kódolás betűinek [1]

Betű	**Morze jel**
A	.-
D	-..
F	..-.
G	--.
V	...-
X	-..-

Az ADFGVX kódolásnál egy 6x6-os táblázatba írják be az ABC betűit és a számokat (7. táblázat). Természetesen a címzettnek és a feladónak előtte meg kell állapodnia a kitöltés rendszerében.

Az üzenet kódolásánál először meghatározzák az adott betű sorát, illetve oszlopát és az azokat azonosító betűpárral helyettesítik:

```
nyíltszöveg:      a  t  t  a  c  k  a  t  1  0  p  m
1.kódszöveg:      DV DD DD DV FG FD DV DD AV XG AD GX
```

Ez még nem adna plusz védelmet az eddigi monoalfabetikus kódoláshoz, hiszen betűpárokra alkalmazva a gyakoriságelemzést, könnyen megtörhető lenne a szöveg.

7. táblázat Az ADFGVX kódolás egy lehetséges betűmátrixa [1]

	A	**D**	**F**	**G**	**V**	**X**
A	8	p	3	d	1	n
D	l	t	4	o	a	h
F	7	k	b	c	5	z
G	j	u	6	w	g	m
V	x	s	v	i	r	2
X	9	e	y	0	f	g

Ezért az ADFGVX kódolás további lépésetek tartalmaz. Először is mindkét félnek előzetes meg kellett állapodnia egy kulcsszóban. A feladó a kulcsszó betűinek megfelelő oszlopszámú mátrixba sorfolytonosan beírja a 1. kódszöveg betűit (8. táblázat)

```
1.kódszöveg: DV DD DD DV FG FD DV DD AV XG AD GX
```

8. táblázat A kódszöveg beírása a kulcsszónak megfelelő mátrixba sorfolytonosan [1]

M	**A**	**R**	**K**
D	V	D	D
D	D	D	V
F	G	F	D
D	V	D	D
A	V	X	G
A	D	G	X

Ezután a mátrix oszlopait olyan sorrendbe rakjuk, hogy a kulcsszó betűi ABC sorrendben legyenek (9. táblázat).

9. táblázat Az ADFGVX kódolás egy lehetséges betűmátrixa [1]

A	**K**	**M**	**R**
V	D	D	D
D	V	D	D
G	D	F	F
V	D	D	D
V	G	A	X
D	X	A	G

Ebből a mátrixból oszlopfolytonosan olvassák ki a szöveget, ami már a rádión átküldendő titkosított üzenet:

```
Titkosított üzenet: VD GV VD DV DD GX DD FD AA DD FD XG
```

A címzett a kulcsszó birtokában a megkapott üzenetet a megfelelő oszlopszámú mátrixba beírja, az oszlopokat visszarendezi a kulcsszó betűinek megfelelő sorrendbe és soronként 2-2 betűt használva dekódolja a szöveget az általa is ismert 6x6-os betűmátrix segítségével, ahol az első betű a sort, a második betű pedig az oszlopot adja meg, így azonosítva az eredeti betűt.

Lieutenant Georges Painvin 1918. áprilisában, alig egy hónappal a kódolás bevezetése után sikeresen feltörte azt (érdemes megemlíteni a francia oldalról a kiemelt motivációt a német kódolás megtöréséhez az említett időszakban).

A Vigenére kódolás továbbfejlesztett változata

Az első világháború végén amerikai kriptográfusok kidolgoztak egy új módszert, mely a Vigenére-kódoláson alapul, de nem lesznek a titkosított üzenetben azonos karakterláncsorozatok, melyek végük a kódolás sikeres feltöréséhez vezetett.

A ciklikusságmentességet úgy lehet biztosítani, hogy a nyílt szöveggel azonos hosszúságú kulcsszöveget használnak a titkosításkor.

Ebben az esetben nincs kiindulópontja a támadónak, csak a rádión lehallgatott üzenetet ismeri:

```
Kulcsmondat: ????????????????????
Nyíltszöveg: ????????????????????
Kódszöveg:   VHRHMEUZNFQDEZRWXFIDK
```

Ennek ellenére ezzel a módszerrel titkosított szöveg is megfejthető.

A kriptoanalízis arra a feltevésre alapoz ebben az esetben (a feladó ismeretében az üzenet eredeti nyelve azonosítható), hogy a kódszöveg tartalmaz néhány gyakori szót az adott nyelvből (pl. the).

Ezeket a gyakori szavakat helyettesítik be nyíltszöveg különböző pontjain és megnézik a lehallgatott üzenet alapján, hogy mi lehetett a kulcsmondat azon része:

```
Kulcsmondat: CAN????BSJ????YPT????
Nyíltszöveg: THE????THE????THE????
Kódszöveg:   VHRMHEUZNFQDEZRWXFIDK
```

Ha a THE nem megfelelő helyre került, akkor értelmetlen betűsorozatot kapunk (ebben az esetben 3 betűből állót), viszont jó helyre téve a kulcsmondat egy szavának részét kapjuk eredményül, mely alapján a használt kulcsszó kitalálható (próbálgatással több lehetőség esetén)).

Pl. ha az YPT valóban része a kulcsnak, akkor csak három megoldás lehetséges: Apocalyptic, Crypt, Egypt, illetve ezek származékai.

Ha a három szójelöltet behelyettesítjük a kulcsmondatba, akkor pedig megfelelő szó esetén a nyíltszövegből sikerült felfedni egy részben értelemezhető darabot:

```
Kulcsmondat: CAN?????APOCALYPTIC??
Nyíltszöveg: THE?????nqcbeothexg??
Kódszöveg:   VHRMHEUZNFQDEZRWXFIDK

Kulcsmondat: CAN??????????CRYPT????
Nyíltszöveg: THE??????????cithe????
Kódszöveg:   VHRMHEUZNFQDEZRWXFIDK

Kulcsmondat: CAN??????????EGYPT????
Nyíltszöveg: THE??????????ATTHE????
Kódszöveg:   VHRMHEUZNFQDEZRWXFIDK
```

Úgy néz ki, hogy egy országnév értelmezhető eredményt hozott, ezért vélelmezzük, hogy a kulcsmondat országok neveit tartalmazza, így CAN-t kiegészítve CANADA-ra próbát teszünk, mit kapunk a nyílt szövegben:

```
Kulcsmondat::     CANADA??????EGYPT????
Nyíltszöveg: themee??????atthe????
Kódszöveg:   VHRMHEUZNFQDEZRWXFIDK
```

A THEMEE, lehet THEMEETING, ezért erre egészítjük kia nyíltszöveget és megnézzük mit kapunk a kulcsmondatban:

```
Kulcsmondat: CANADABRAZ??EGYPT????
Nyíltszöveg: themeeting??atthe????
Kódszöveg:   VHRMHEUZNFQDEZRWXFIDK
```

BRAZ-t kiegészítve BRAZIL-ra ismét a nyíltszöveg egy darabkáját fedtük fel sikeresen:

```
Kulcsmondat: CANADABRAZILEGYPT????
Nyíltszöveg: themeetingisatthe????
Kódszöveg:   VHRMHEUZNFQDEZRWXFIDK
```

Már csak négy betű ismeretlen a kulcsmondatból és a nyílszövegből. A 4 betűs országneveket próbálgatva behelyettesíteni a kulcsmondat adott helyére végül megkapjuk a teljes nyílt szöveget.

```
Kulcsmondat: CANADABRAZILEGYPTCUBA
Nyíltszöveg: themeetingisatthedock
Kódszöveg:   VHRMHEUZNFQDEZRWXFIDK
```

Láthattunk, hogy ugyan megszüntettük a karakterláncok ismétlődését a kódolt szövegben, mégis vissza tudtuk fejteni, mert mind a nyílszöveg, mind a kulcsmondat értelmes szavakat tartalmazott, így szódarabokat azonosítva és azokat sikeresen kiegészítve egymást fedte fel a kulcsmondat és a nyílt szöveg, a kódolt üzenet birtokában és ismerve, hogy Vigenére kódolást használtak.

One Time Pad

1918-ban a kriptográfusok az előző módszer törhetősége után strukturálatlan, véletlenszerűen képzett kulcsokkal próbálkoztak és végül egy feltörhetetlen kódhoz jutottak el (pedig korábban írtam, hogy minden rendszer feltörhető, ez is, majd látni fogjuk, mégis bizonytalanok leszünk az eredményben).

Az amerikaiak összeállítottak egy több száz lapból álló tömböt, melynek minden egyes lapja más-más random kulcsot tartalmazott. Ebből a tömbből két példányt készítettek, egyet a feladónak és egyet a későbbi címzettnek adtak. A feladó a kódoláshoz a Vigenére-táblát és az első lapon olvasható kulcsot használta (10. táblázat).

10. táblázat Az One Time Pad egy lapjának elképzelt tartalma [1]

P	L	M	O	E
Z	Q	K	J	Z
L	R	T	E	A
V	C	R	C	B
Y	N	N	R	B

```
Kulcsmondat: PLMOEZQKJZLRTEAVCRCBY
Nyíltszöveg: attackthevalleyatdawn
Kódszöveg:   PEFOGJJRNULCEIYVVUCXL
```

A címzett a visszafejtéshez szintén a Vigenére-táblát és az első lapot használta.

Ezután az első lapot elégették és a következő üzenetváltáshoz a következő lapon szereplő kulcsot használták fel.

Mivel minden kulcsot csak egyszer használtak, innen ered a neve. A mai kriptorendszerek egy része szintén egyszer használatos kulcsokkal dolgozik, így növelve a biztonságot.

Mivel a kulcsban nincs ismétlődés, ezért a Babbage és Kasiski által kidolgozott módszer nem alkalmazható a megfejtéshez.

Aki lehallgatta a rádióadást, hozzájuk a kódszöveghez.

A gyakran előforduló szavakat különböző helyre illesztve mindig értelmetlen betűsorozatot kap ezért nem tudja eldönteni, hogy a próbaszó jó helyen van-e.

A végére a brure force algoritmus marad, mely esetében minden helyre minden betűt behelyettesítve próbáljuk meg a szöveg megfejtését.

Ez a módszer működik, de így minden egyes értelmes szöveg is előáll azon a karakterlánc hosszon, így nem lehet eldönteni, melyik a valódi szöveg:

```
Kulcsmondat: PLMOEZQKJZLRTEAVCRCBY
Nyíltszöveg: defendthehillatsunset
Kódszöveg:   PEFOGJJRNULCEIYVVUCXL
```

Ebben rejlik eme kódolás erőssége.

Abban az esetben, ha a lehallgató személy rendelkezés elég információval az üzenetváltókról, akkor a lehetséges válaszokból ki tudja találni, melyik lehet az eredeti üzenet az összes értelmes szöveg közül.

Nihilista kódolás

1880-ban használták az orosz nihilisták a cári rendszer elleni terroresemények megszervezéséhez, melyben Polybios-féle 5x5-ös mátrixot használtak kulcsszóval (pl. ZEBRAS), illetve még egy kulcsot (pl. NARODNIK) a kódolás során így nehezítve a megfejtést ezek hiányában.

A kulcstábla elkészítéséhez először a kulcsszó betűivel kell kitölteni a mezőket (az ismétlődő betűket csak egyszer leírva), majd az üresen maradt helyeket az ábécé többi betűjével kell kitölteni (az I-t és a J-t egy betűként kezelik).

A kulcsszó használata biztosítja a mátrix bármikor könnyen kitölthetőségét, mivel nem kell megjegyezni minden egyes betű helyét, az következik a kulcsszó beírása után.

A kulcsszót írhatjuk a tábla legfelső sorába, balról jobbra, vagy valamelyik más elrendezés szerint, például a bal felső sarokból induló spirálban, amely középen végződik.

A kulcsszó az 5×5-ös tábla kitöltési módjával együtt alkotja magát a kódot.

11. táblázat A Nihilista kódolásnál használt mártix (példa)

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

A nyíltszöveg titkosításakor első körben az üzenet betűinek megfelelő számpárokat képzik a mátrix alapján (sor és oszlop):

```
Nyíltszöveg:      W  I  N  T  E  R  P  A  L  A  C  E
Kódolt szöveg:    53 32 41 45 12 14 43 15 34 15 22 12
```

Ezután a kulcs (NARODNIK) betűinek és képzik a mátrix alapján a számpárosát, szükségszerűen ismételve, ha a titkosítandó üzenet nála hosszabb:

```
Kulcsszó:     41 15 14 42 23 41 32 33 41 15 14 42
```

Végül az adott pozícióban kapott számpárokat összeadják és ezt küldik el a címzettnek:

```
Kódolt üzenet: 94 47 55 87 35 55 75 48 75 30 36 56
```

A címzett első körben kivonja a kulcsként használt szó betűinek mátrix által meghatározott értékét a kapott üzenet értékeiből sorra, majd a maradék értékhez tartozó számpárok alapján kikeresi a mártixból, hogy adott sorhoz és oszlophoz tartozó cellában milyen betű szerepel, így visszakapja az eredeti üzenet betűit.

Bifid kód

Felix Delastelle 1901-ben alkalmazta először ezt a kódolási módszert.

Első lépés: Egy Polybios-féle mátrixba írt ABC alapján kódoljuk az üzenetet (itt is haszánlhatunk kulcsszót a könnyebb megjegyezhetőségre a kitöltéssel kapcsolatban):

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

A mártix alapján kódoljuk az üzenetet de úgy, hogy az üzenet betűinek megfelelő sor értéket az első sorba, az oszlopértéket alá a második sorba írjuk:

```
U Z E N E T
4 1 3 2 3 5
4 5 5 3 5 1
```

Az így kapott két soros számsort sorfolytonosan egymásután fűzzük:

```
Sorfolytonosan: 4 1 3 2 3 5 4 5 5 3 5 1
```

Majd ebből számpárokat képezve a mátrix alapján kódoljuk újra és megkapjuk az elküldendő üzenetet:

```
Új kódolás:  41 32 35 45 53 51
Küldendő:    F  O  E  M  Y  T
```

Playfair kódolás

1854-ben Charles Wheatstone találta fel, de legfőbb támogatójának, Lord Playfairnek a nevét viseli. Amódszer betűpárokat (*hasonlóan a Grannd Chiffre-hez*) kódol betűket helyettesítések helyett.

A Playfair kódolás egy 5×5-ös táblát használ, amely tartalmaz egy kulcsszót. A kulcstábla elkészítéséhez először a kulcsszó betűivel kell kitölteni a mezőket (az ismétlődő betűket csak egyszer leírva), majd az üresen maradt helyeket az ábécé többi betűjével kell kitölteni (ez vagy a Q betű kihagyásával, vagy pedig az I-t és a J-t egy betűként kezelve.) A kulcsszót írhatjuk a tábla legfelső sorába, balról jobbra, vagy valamelyik más elrendezés szerint, például a bal felső sarokból induló spirálban, amely középen végződik. A kulcsszó az 5×5-ös tábla kitöltési módjával együtt alkotja magát a kódot.

A szöveg kódolásához az eredeti szöveget betűpáronként vesszük és alkalmazuk az alábbi 4 szabály valamelyikét:

- ha a betűpár mindkét betűje ugyanabban a sorban jelenik meg a kulcstáblán, akkor a tőlük közvetlenül jobbra állóval kell helyettesíteni őket (ha történetesen az egyik betű a sor jobb szélén van, akkor a sor bal szélén álló betűvel kell helyettesíteni):

```
1. * * * * *
2. * A P Z F
3. * * * * *
4. * * * * *
5. * * * * *
```

Itt AZ betűpárból PF lesz.

- Ha betűpár pár mindkét betűje ugyanabban az oszlopban jelenik meg a kulcstáblán, akkor közvetlenül az alattuk állóval kell helyettesíteni őket (ha az egyik betű az oszlop alján van, akkor az oszlop tetején álló betűvel kell helyettesíteni).

```
1. * * A * *
2. * * B * *
3. * * * * *
4. * * Z * *
5. * * Y * *
```

Itt az AZ betűpárból BY lesz

- Ha a betűpárpár betűi nincsenek sem egy sorban, sem egy oszlopban (ez fordul elő a legtöbb esetben), akkor tekintsük azt a kulcstábla mezőiből felépülő téglalapot, amelynek a két betű a két szemközti csúcsa. A betűket a saját sorukban, a téglalap másik csúcsánál lévő betűkkel helyettesítjük. Pl. AZ kódolása:

```
1. B * * A *
2. * * * * *
3. * * * * *
4. Z * * X *
5. * * * * *
```

Itt az AZ betűpárból BX lesz

- Ha az egyik pár mindkét eleme ugyanaz a betű, vagy már csak egy betű maradt az utolsó párba, akkor írjunk egy "X"-et az első betű után, és ezt az új párt kódoljuk az előző 3 szabály megfelelőjével.

Katonai alkalmazása volt jelentős. A második búr háborúban és az I. világháborúban a brit hadsereg használta, az ausztrálok pedig a II. világháború alatt. Az ok egyszerű, a Playfair gyorsan és bármiféle különleges felszerelés nélkül használható, viszont mire az ellenség feltöri a kódot, a szöveg tartalma egy csatában már érvényét veszti.

Tehát a Playfairt sokkal nehezebb feltörni, mivel a gyakoriságelemzést 600 betűpárra kell elvégezni a 26 betű helyett egy angol szöveg esetén, 1600 betűpárra a 40 betű helyett magyar nyelvűnél. Látható, hogy ugyan nem lehetetlen, de tovább tart a törés, illetve hosszabb kódolt szöveg szükséges hozzá, hogy a nyelvi jellemzők a betűpárok esetén hangsúlyosabbak legyenek.

Egy másik módszer a Playfair kódolás megtörésére az ún. *shotgun hillclimbing* módszer, amely egy véletlenszerűen kitöltött négyzettel próbálkozik, amelyen kisebb változtatásokat eszközölve (betűk, sorok felcserélése vagy az egész négyzet tükrözése) rendre ellenőrizzük, hogy a kapott szöveg jobban hasonlít-e egy normál szöveghez, mint az előző. Ha javulást látunk, akkor az utolsó változtatást követve igyekszünk még pontosabb megoldást találni addig, amíg, még ha nem is teljesen azonos, de már értelmezhető szöveget kapunk.

A négy négyzet kódolás

Felix Delastelle találta fel 1901.ben és a Playfair kódoláshoz hasonlóan betűpárokat kódolunk a segítségével. Négy darab 5×5-ös mátrixot használ, amelyek egy nagy négyzetet alkotnak. Mindegyik négyzet tartalmazza az angol ábécé betűit, rendszerint vagy a Q-t elhagyva, vagy pedig az I-t és a J-t egy betűnek tekintve, hogy az amúgy 26 betűből álló angol ábécé betűi elférjenek. Általában a bal felső és jobb alsó négyzetek az eredeti szöveghez tartozó négyzetek, melyek egy-egy standard ábécét tartalmaznak. A jobb felső és bal alsó négyzetek pedig a kódolt szöveghez tartoznak, és az ábécének egy-egy kevert sorrendjét tartalmazzák két különböző kulcsszó alapján kitöltve (pl. PÉLDA és KULCSSZO):

```
a b c d e     P E L D A
f g h i j     B C F G H
k l m n o     I J K M N
p r s t u     O R S T U
v w x y z     V W X Y Z

K U L C S     a b c d e
Z O A B D     f g h i j
E F G H I     k l m n o
J M N P R     p r s t u
T V X Y Z     v w x y z
```

.

Egy üzenet kódolásához az alábbi lépéseket kell végrehajtani:

- a szöveg betűpárokra bontása
- a- betűpár első tagjának megkeresése a bal felső mátrixban
- a betűpár második tagjának megkeresése a jobb alsó mátrixban
- a kódolt betűpár első tagját az eredeti betűpár első tagjának sorában, második tagjának oszlopában találjuk
- a kódolt betűpár második tagját az eredeti betűpár második tagjának sorában, első tagjának oszlopában találjuk

Tehát a kódoláshoz meg kell találnunk annak a téglalapnak a két másik csúcsát, melynek két csúcsát már ismerjük, s ezekben a csúcsokban található betűk adják a kódolt szöveget.

a b c d e	P E L D A
f g ***h*** i j	B C F G ***H***
k l m n o	I J K M N
p r s t u	O R S T U
v w x y z	V W X Y Z
K U ***L*** C S	a b c d ***e***
Z O A B D	f g h i j
E F G H I	k l m n o
J M N P R	p r s t u
T V X Y Z	v w x y z

Itt a HE betűpárból HL lesz

A dekódolás ugyanezen az elven működik, csak fordítva: a betűpár tagjait nem az eredeti szöveg, hanem a kódolt szövegnek megfelelő mátrixokban helyezzük el, és így keressük a kialakuló téglalap csúcsait.

A Playfair-rejtjeltől eltérően a négy négyzet-rejtjellel fordított betűpárokat kódolva nem kapunk fordítottakat (például a Playfair-rejtjellel AB BA valamilyen XY YX-re lesz kódolva, de a négy négyzettel nem). Mindez persze csak akkor áll fenn, ha a két kulcsszó különböző. A négy négyzet erősebb kódolási rendszer a Playfairnél, azonban lassabb is, a két kulcs használata és a kódoló/dekodoló lap előkészítése miatt.

A Playfair terjedt el szélesebb körben.

Jefferson henger

Thomas Jefferson az USA elnöke és a Függetlenség nyilatkozat egyik megfogalmazója fejleszti ki. 20-30 tárcsa van egy tengelyen és inden tárcsára más sorrendben írták fel az ABC betűit. Szabványosítva van természetesen, hogy az 1. számú tárcsán milyen sorrendben, a 2. számú tárcsán milyen sorrendben, stb., különben nem használhatná a címzett az eszközt a dekódoláshoz.

A tárcsákat a feladó és a címzett a megbeszélt sorrendben felfűzik a hengerre,

A feladó beállítja a küldendő üzenetnek megfelelően a tárcsákat és a megbeszélt sorból (alatta, vagy felette mennyivel) kiolvasott üzenetet küldik el (34. ábra).

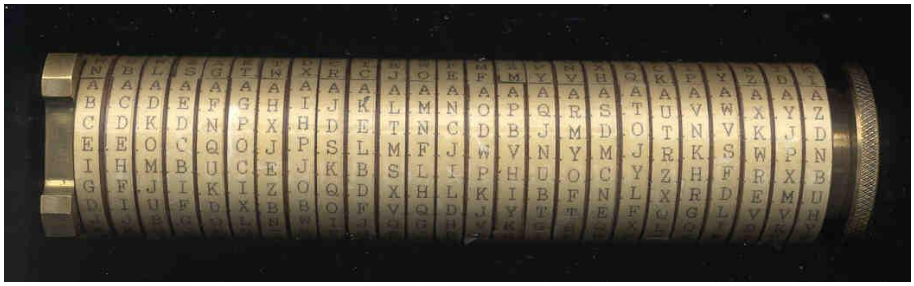
34. ábra Jefferson henger [1]

M-94

Jefferson idejében nem terjedt el kellőképpen a találmánya, viszont egy évszázaddal később Étienne Bazeries a „Nagy Kód” feltörője találta fel „újra”.

Az M-94-et ennek alapján fejlesztették ki 1922-ben és használták a II. VH elejéig.

25 db. alumínium tárcsát tartalmaz, és a Jefferson-féle módszert alkalmazzák az üzenet titkosításához (35. ábra).

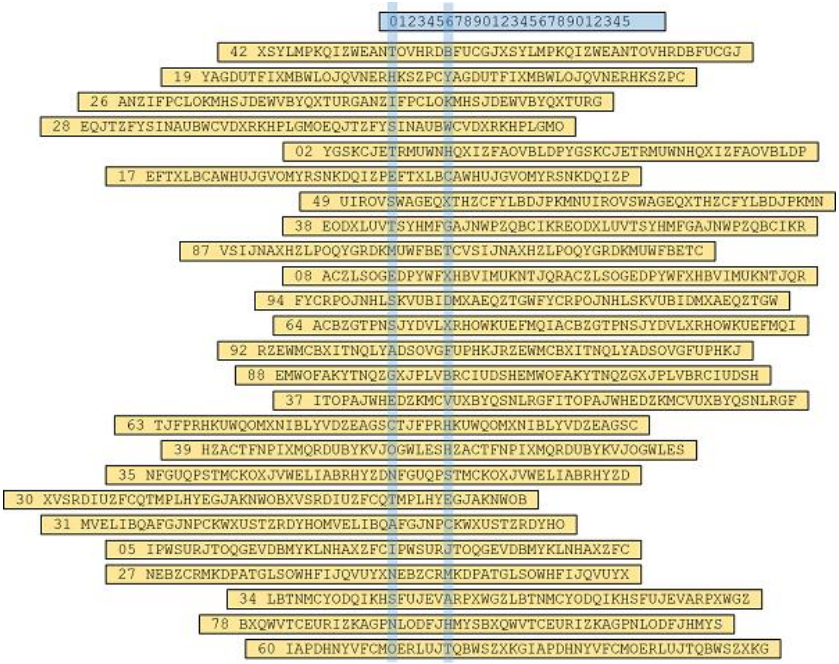


35. ábra M-94 [1]

M-138-A

100 pálcikát tartalmaz, melyek véletlenszerűen tartalmazzák az ABS betűit (megint szabványosítva, az összes 1-es számú pálcika azonos sorrendben), melyből 30-at használnak 1-1 üzenet titkosításához. Természetesen a címzettnek és a feladónak előre meg kellett állapodnia, mely pálcikákat és milyen sorrendben fűzik bele az eszközbe.

A középen lévő függőleges vonalhoz igazították az üzenet betűit egymás alá, majd a címzettel korábban egyeztetett, jobbra, vagy balra adott távolságra lévő oszlop betűit küldték el üzenetként (36. ábra).



36. ábra M-138-A [3]

Visszafejtésnél azonos pálcikákat, azonos sorrendben befűzbe az eszközbe úgy mozgatjuk őtek, hogy a középső vonal alatt legyenek a megkapott üzenet betűi és a megbeszélt oszloppal eltolt helyen találjuk az eredeti üzenetet.

1944-ig használták, amikor a németek feltörték ezt a titkosítási módszert.

A II. Világháború, a kódológépek elterjedése

A II. Világháború előtt jellemzően kézzel történt a titkosítás és visszafejtés a korábbiakban ismertetett módon. Ugyan voltak törekvések ennek gépesítésére, de egyészen az Enigma megjelenéséig nem jártak sikerrel.

Enigma

Az Enigma megváltoztatta a játékszabályokat, sokkal gyorsabban és a legfontosabb, pontosabban lehetet elvégezni a titkosítást és a visszafejtést a segítségével.

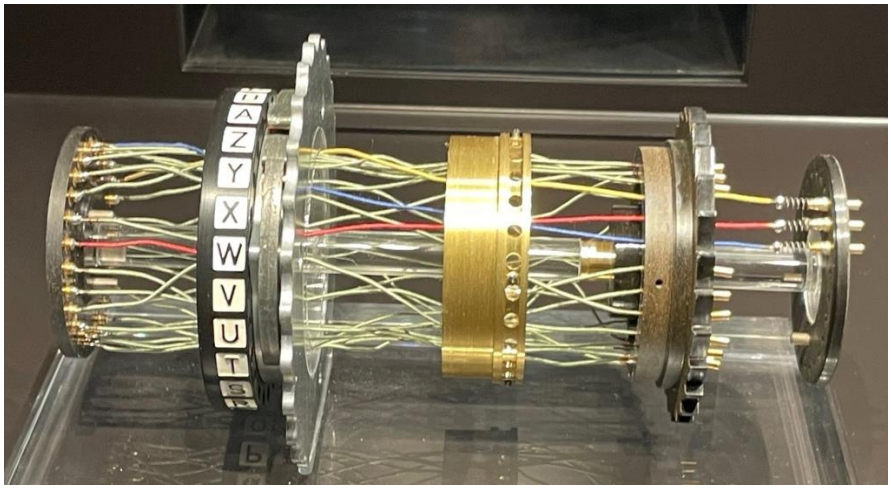
Arthur Scherbius német feltaláló szerkesztette meg az Enigmát 1918-ban, melyet ebben az időszakban a cégek kezdtek el használni az ipari kémkedést megakadályozandó (már volt ennek fontosságáról szó korábban).

Három fő egységből áll, amelyeket vezetékek kötnek össze: egy *billentyűzet* a nyílt szöveg (visszafejtésnél a kódolt szöveg) betűinek bevitelére; egy *keverőegység (alapból 3 keverőtárcsával),* amely a nyílt szöveg betűit a kódszöveg megfelelő betűivé alakítja; és egy *kijelző panel,* amelyen adott betűhöz tartozó lámpa felvillanása jelzi a bevitt betű kódolt változatát (37. ábra). Tehát betűt betűre kódol. Ilyennel már találkoztunk, de most a kódolás összetettsége, ami változik, így növelve a biztonságot.

37. ábra Enigma [1]

A nyílt szöveg adott betűjének sifrírozásához a gépkezelő a billentyűzeten leüti a megfelelő billentyűt. Ennek hatására elektromos impulzust fut végig a központi keverőegységen keresztül, ahonnan a jel kijut a kijelzőtáblára és megvilágítja a megfelelő betű lámpáját. A keverőtárcsára az ABC 26 betűjét (esetleg 1-26-ig a számokat) írták fel (kell a beállításhoz), és 26 vezeték fut benne egyik oldalról a másikra, összekötve ezáltal a bal és jobb oldalon lévő 26-26 érintkezőt. A keverőtárcsán van egy nagyobb átmérőjű fogaskerékre hasonlító gyűrű, mely a gépen belüli forgatását/adott pozícióba állítását segíti. A billentyűzetről huszonhat vezeték fut a keverőbe, a keverőtárcsák belső huzalozása szerint lép ki a másik oldalon a belépő elektromos impulzus a

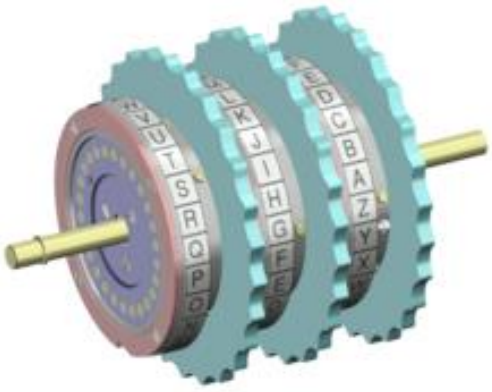
szomszédos tárcsába, majd ezeken végig haladva szintén huszonhat ponton lép ki. A nyílt szöveg betűinek sifrírozását a keverőtárcsa belső huzalozása határozza meg (nem a szemben lévő érintkezőket kötik össze a két oldalon, hanem egy szabványosított módon összekeverve őket), mely azonos számmal ellátott tárcsa esetén azonos (38. ábra). Érdemes megnézni, hogy a jobb oldali 26 érintkező össze van kötve egy kevert módon a bal oldali 26 érintkezővel. A gyártás során fontos volt figyelni arra, hogy az azonos számmal ellátott keverőtárcsában azonos legyen a huzalok összekeverése is, hogy két külön gép azonos számmal ellátott keverőtárcsával azonos eredményt adjon. Kicsit úgy kell elképzelni, hogy amikor veszünk egy számológépet (tudom- már inkább a telefonon lévő applikációt használjuk, de ott is ez a helyzet), akkor mindegy, hogy melyiket vesszük, azonos számításnál azonos eredményt kell adnia, ahogy a különböző telefonokre letöltött különböző számológép-applikációknak is.



38. ábra Keverőtárcsa robbantott képe [3]

Annak érdekében, hogy ugyanazt a betűt leütve ne mindig ugyanahhoz a betűhöz tartozó lámpához jusson el az impulzus és kapcsolja fel, annak érdekében Scherbius megoldotta, hogy a 3 keverőtárcsából az első (jobbról nézve) minden billentyűleütés után elfordul 1/26-od fordulattal (a palástot nézve egy betűvel). Ennek köszönhetően ugyanannak a betűnek a leütésével generált impulzus az elfordult tárcsa belső huzalozása miatt már nem ugyanazon a ponton lép ki a másik oldalán és fut tovább a következő tárcsán.

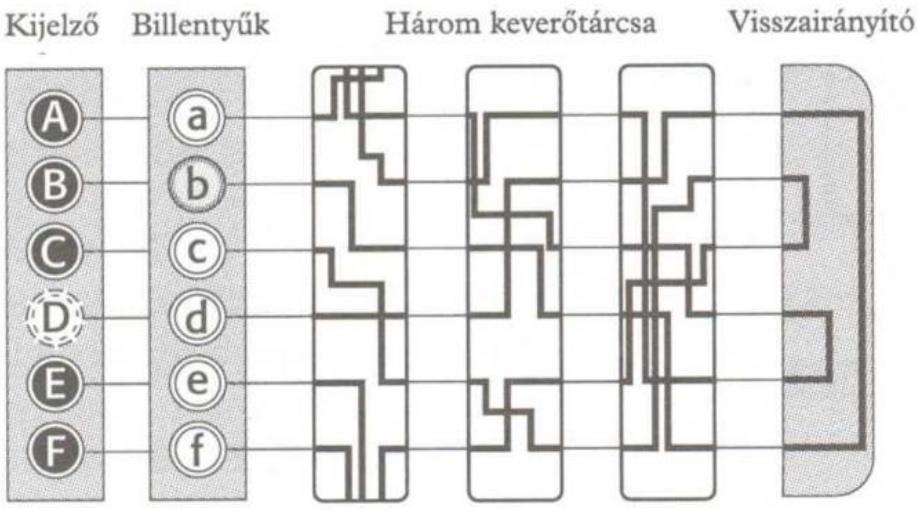
Ha belegondolunk, akkor 26 leütés után ugyanabba pozícióba kerülne a tárcsa, tehát ugyanannak a betűnek azonos lenne a kódolt változata, így 26 betűnként gyakoriságelemzés végezve megtörhetnénk az üzeneteket. Nem hiába gyűjtöttek a kriptográfusok össze több évszázadnyi tudást, Scherbius sem véletlenül használt 3 tárcsát a keverőegységében (39. ábra)



39. ábra Keverőtárcsák a keverőegységben [3]

A második tárcsa akkor fordult el 1/26-ot fordulattal, amikor az első keverőtárcsa már egy teljes kört megtett. A harmadik tárcsa pedig csak a második tárcsa teljes körbefordulása után fordul el egy betűnyit.

Ezáltal 26x26x26=17.576 különböző pozíciót vehet fel, tehát a gyakoriságelemzés a szövegből minden 17.576-ik betű kivételére lenne alkalmazható, de az üzenetek eleve nem voltak ilyen hosszúak.



40. ábra A visszairányító szerepe (egyszerűsített ábra) [1]

Ahhoz, hogy a kódolt üzenetet vissza tudjuk fejteni, ahhoz először az Enigmát azonos beállítással kell indítani mind a feladónak, mind a címzettnek. A visszairányító résznek köszönhetjük azt, hogy azonos beállítás mellett a kódolt betű visszavezet az eredeti betűre. Nézzük meg az áttekinthetőség miatt egyszerűsített, csak 6 betűt tartalmazó ábrát 40. ábrát és a billentyűzeten a „B" betűtől induló vonalat kövessük végig keresztül a tárcsákon a visszairányítók keresztül a kijelzőig. Ha jól csináljuk, a „D" betűhöz tartozó lámpa világít, ezt küldték el a címzettnek. Most próbáljuk meg címzettként dekódolni a „D" betűt. A „D" betűtől indulva, végig haladva a tárcsákon és visszairányítón eljutunk a „B" lámpához, mely a mi Enigmánkon világít majd. Ez volt az eredeti betű. Azonos beállításnak és a visszairányítónak köszönhetően. Másik beállításnál egy másik betűhöz jutna el így az elektromos impulzus.

Az alapbeállításokat általában egy kódkönyv határozza meg, amely minden egyes napra meghatározza a kulcsot, és amely az adott kommunikációs hálózaton belül mindenki által elérhető. A kódkönyv szétosztása idő- és munkaigényes tevékenység, de mivel egy napra csak egy kulcs kell, megoldható, hogy egy huszonnyolc kulcsot tartalmazó kódkönyvet négyhetente kiosszanak. Az alapbeállítás tartalmazza, hogy melyik tárcsa, melyik kiindulópozícióból indul (melyik betű lesz a tárcsán legfelül, hogy forgassuk el kezdés előtt.)

Amennyiben rendelkezünk egy Enigmával és sikeresen lehallgattunk egy vele kódolt üzenetet, akkor a brute force metódust követve (minden beállítást kipróbálva) maximum 17.576 beállítást kellene kipróbálnunk, mely ugyan sokáig tartana, de végül eredményre vezetne.

Viszont már említettünk, hogy a kriptográfusok képben voltak a korábbi eljárások megtörését tekintve és egy ilyen gyenge pontot nem hagytak a készülékben. A keverőtárcsákat úgy készítették el, hogy a 3 helyből bármelyikbe betehetők legyenek, így a 6 keverőtárcsát 6 féle sorrendben lehet elhelyezni (a napi alapbeállítás azt is tartalmazta, melyik tárcsa hova kerüljön).

Ezáltal a végigpróbálandó beállítások száma brute force esetén: 6x17.576=105.456.

Egy kapcsolótáblát is illesztettek az Enigma elejére (41. ábra), melyhez 6 vezetéket adva 12 betűt lehetett egymással felcseréni. Ezzel azt érték el, hogy pl. a „B” és „A” betűket felcserélve, ha a billentyűzeten a „B” betűt nyomták le, az elektromos impulzus úgy haladt át a gépen, mintha az „A” betűtől indult volna és ennek megfelelően kódolta a „B” betűt. Dekódolásnál ugyanazokat a betűket kellett felcserélni, hogy a folyamat végén az eredeti üzenetet kapjuk vissza.

41. ábra Kapcsolótábla (saját kép)

Az aznapi alapbeállítás azt is tartalmazta, hogy melyik betűt, melyik betűvel kell felcserélni a vezetékek segítségével.

Egy lehetséges napi kulcs:
Kapcsolótábla beállítása: A/L-P/R-T/D-B/W-K/F-O/Y
Keverők sorrendje: *3-1-2*
Keverők beállítása: *V*-G-R

Most tekintsük át, hogy hány féle beállítása is létezhet az Enigmának:
Keverőtárcsák beállítása: 26x26x26 =17576
Keverőtárcsák sorrendje: 123,132,213,231,312,321=6
Kapcsolótábla: Egy huszonhat betűs ábécé hat betűpárja 100.391.791.500 módon cserélhető fel
Összesen 17 576 x 6 x 100 391 791 500 = kb. l0 000 000 000 000 000 = 10^{15}

Tehát az üzenet sikeres feltöréséhez 10^{15} beállítást kell kipróbálnunk, mely a mostani számítógépek esetén is eltart egy darabig.

Az első világháborút követő években az angolok, 40-es szoba kriptográfusai továbbra is figyelemmel kísérték a németek kommunikációját.
1926-ban kezdtek olyan titkosított üzeneteket fogni, amelyek teljesen eltértek a korábbiaktól: elkezdték használni a németek az Enigmát.
Az amerikaiak és a franciák is megpróbálták feltörni az Enigma kódját, de végül feladták.

Hans- Thilo Schmidt 1931. november 8-án tízezer márkáért eladott egy francia hírszerzőnek két fotókópiát:

- "Gebrauchsanweisung für die Chiffriermaschine Enigma"
- "Schlüsselanleitung für die Chiffriermaschine Enigma"

címűt.

Ezek az Enigma használati útmutatói voltak, nem tartalmazták a keverőtárcsák belső huzalozási módját, de olyan utalásokat azonban igen, amelyek alapján a huzalozás kikövetkeztethetővé vált.

Schmidt árulásának köszönhetően a szövetségesek most már megtudták építeni a német katonai Enigma pontos mását.

A feltöréséhez viszont továbbra is 10^{15} beállítást kell kipróbálni.

Az Enigma kriptográfiai célokra történő előterjesztésénél így fogalmaztak:

"A titkosítási módszer biztonságának értékelésekor feltételeztük, hogy a gép az ellenség rendelkezésére áll."

Háborús helyzetben nagyon sok üzenetváltás történik egy nap és az Enigma esetén mindegyiket ugyanazzal az alapbeállítással teszik a Wehrmacht egységei és egy másikkal a Luftwaffe egységei. Túl sok üzenet azonos kulccsal már korábban sem erősítette a titkosított üzeneteket, ezért a németek bevezették, a napi beállítással minden üzenettel egy-egy új *üzenetkulcsot* küldtek. Az üzenetkulcs ugyanazt tartalmazta, hogy az üzenet további olvasásához, milye módosítást kell végezni az Enigmán, tehát onnantól kezdve már nem a napi beállítást használták. A kapcsolótábla-beállítást és keverőtárcsa-sorrendet nem változtatták meg (több időt vett volna el), de a keverőtárcsákat másik pozícióba forgatták, mégpedig az üzenetkulcsban közölt pozícióba. Pl. aznap a keverőtárcsákat C-T-W pozícióba kellett forgatni, akkor ezzel a beállítással küldte el az új pl. T-F-S pozíciókat jelezve, hogy mielőtt a feladó folytatja az üzenetküldést, előtt ebbe a pozícióba állítja a tárcsákat. Így a címzett az aznapi beállítással tudja dekódolni és ő is módosít a további titkosított üzenetek visszafejtéséhez az Enigmáján. Ezt mindenki csak 1x tette meg, az üzenet elején és minden üzenetnél más-más üzenetkulcsot (beállítást) átküldve.

A németek annak érdekében, hogy kiküszöbölték a rádió interferenciából adódó hibákat, az üzenetkulcsot duplázva küldték el, tehát az előző példa alapján az aznapi beállítással kódolva átküldték a TFSTFS üzenetet és csak utána állították át a tárcsákat ennek megfelelő pozícióba.

Vagyis az üzenet első hat betűje mindig az aznapi beállítással volt kódolva és csak a 7. betűtől kezdve változtak meg a beállítások a kommunikációban résztvevő Enigmákban.

Rejewski-féle törés

Rejewski lengyel matematikus a német üzenetkulcs duplázás (pl. T-F-S-T-F-S) módszerét használta fel arra, hogy megtörje az Enigmát. A rádión lehallgatott üzenetek első hat betűje az aznapi beállítással van titkosítva. Nézzünk pár üzenetkezdetet:

1. Üzenet L O K R G M
2. Üzenet M V T X Z E
3. Üzenet J K T M P E
4. Üzenet D V Y P Z X

Az első és a negyedik betű mind a négy esetben ugyanannak a betűnek, nevezetesen az üzenetkulcs első betűjének a sifréje. Ugyanazt a betűt jelöli a második és az ötödik, illetve a harmadik és a hatodik helyen álló sifre is. Elég sok üzenetet lehallgatva el tudott készíteni 3

táblázatot az 1. és 4., a 2. és 5., valamint a 3. és 6. betűkre vonatkozóan úgy, hogy amikor pl az első betű „A” volt, akkor a 4. „F”, stb.

```
1. betű      ABCDEFGHIJKLMNOPQRSTUVWXYZ
4. Betű FQHPLWOGBMVRXUVCZITNJEASDK
```

Ezekben a táblázatokban betűláncokat kezdett el keresni pl.:

```
A - F - W - A                                3 láncszem
B - Q - Z - K - V - E -L -R -I -B            9 láncszem
C - H - G - O - Y - D - P - C                7 láncszem
.   .     .     .     .     .     .     .
J - M - X - S - T - N - U - J                7 láncszem
```

Azt figyelte meg, hogy a láncok hossza egyedi minden beállításra, a keverőtárcsák sorrendjétől és a kiinduló pozíciójuktól függ, de a kapcsolótábla beállítása nincs rá hatással.

Ez azt jelenti, hogy a 6x17.576 beállításra kell csak koncentrálni és a kapcsolótábla által biztosított *100.391.791.500* lehetőséggel nem kell foglalkozni, tehát 100 milliárddal csökkentette a brute force eljárás által kipróbálandó beállításokat.

Rejewski az Enigma 6x17.576=105.456 beállításához egy könyvben felvezette, melyik beállítás, milyen lánchossz tulajdonságokkal bír.

Ezután ha rendelkezésére állt elegendő üzenet az aznapi beállításokkal ahhoz, hogy meg tudja határozni a láncok hosszát, a könyvből kikeresve meghatározta az aznapi beállítást és dekódolta az üzenetet.

Az üzenet tartalmazta a kapcsolótáblából adódó betűcseréket pl.: alliveinbelrin, de az, hogy melyik betűt, melyikre cserélték ki, már könnyen meghatározható volt.

Rejewski tehát sikeresen feltörte az Enigmát mégpedig 1934-ben.

1938. decemberében a németek növelték az Enigma biztonságát azáltal, hogy a már 3 keverőtárcsa mellé két-két új keverőtárcsát adtak, így már 5 tárcsából lehetett 3-at behelyezni a készülékbe, így a tárcsák hatvanféle sorrendben voltak elhelyezhetők.

Ez 60x17.576=1.054.560 beállításhoz vezetett.

A következő hónapban a kapcsolótábla vezetékeinek számát hatról tízre emelték, így keverőtárcsákba történő jelbevitel előtt már húsz betű felcserélésével kellett számolni.

Az elméletileg lehetséges kulcsok száma 159.000.000.000.000.000.000-ra ($159*10^{18}$) nőtt.

Megjegyezzük, hogy a Rejewski törés továbbra is működik, de már millió beállítást kell végignézni a könyvben, melynek vastagsága a tízszeresére nőtt, valamint a 26 betűből 20 felcserélése már a visszafejtett üzenet esetében is komolyabb kihívást jelent, mivel rövidebb értelmes szakaszok találhatók majd a szövegben.

Alan Turing-féle törés

Alan Turing korának nagy matematikusa, a Turin-gép és Turing-teszt kiötlője.

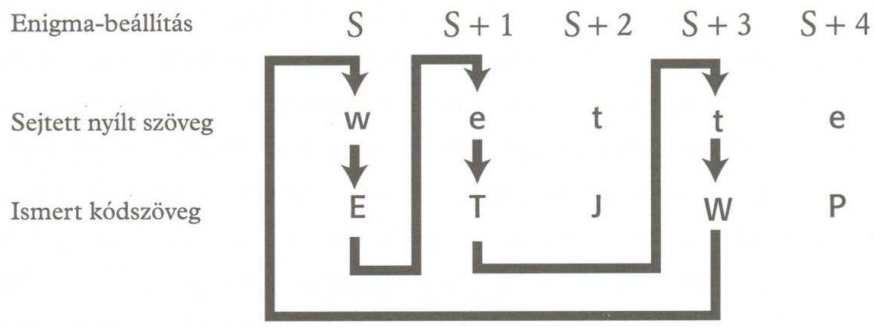
A Bletchley parkban dolgozott több kollégájával az Enigma feltörésén és arra összpontosított, hogy készüljenek fel arra az esetre, ha Németország megváltoztatja az üzenetku1csküldés módszerét. A brit kriptográfusok úgy gondolták, hamarosan a németeknek is feltűnik az üzenetkulcs kétszeri elküldésének problémája, és utasítják az Enigma csak 1x küldjék el az üzenetkulcsot. Ez meg is történt 1940. május 1-én, így a britek 1940. augusztus 8-ig nem tudták megfejteni az üzeneteket (erről a küzdelemről szól a Kódjátszma c. film is).

Turing rájött, hogy az elküldés időpontja és a feladó ismeretében időnként - legalábbis részben - előre meg tudja mondani, mit tartalmaz az eredeti. A gyakorlat például azt mutatta, hogy a németek mindennap valamivel este hat után időjárás-jelentést küldenek. Ezáltal egy l8.05-kor lehallgatott üzenet valószínűleg tartalmazta a Wetter szót. Az üzenetek szigorú szerkezete és a német nyelvtani szabályoz jól behatárolhatóvá teszik, hogy ezekben a kódszövegekben hol helyezkedik el a Wetter szó.

Ahhoz viszont, hogy az ismert tartalom pl. ETJWPX, milyen beállításnál adja vissza a Wetter szót, $159*10^{18}$ beállítást kellene végigpróbálnia.

Turing úgy gondolkodott, hogy ha például talál valamit támpontot, aminek semmi köze a kapcsolótáblához, akkor a fennmaradó 1.054.560 beállítás (60x17.576)) végigpróbálása már nem lehetetlen, a helyes keverőtárcsa-beállításból pedig kikövetkeztethető a kapcsolótábla 20 betűs cseréje.

Turing is betűláncokkal kezdett el foglalkozni, de a hurkok egy-egy támponton belül a kódszöveg és a nyílt szöveg betűit kapcsolták össze (42. ábra).



42. ábra Kapcsolótábla (saját kép)

A beállítások vizsgálatára annyi enigmát használt, amilyen hosszú volt a lánc. A lánc azt is meghatározta, hogy az első enigmához képest az összes többe mennyivel elforgatott pozícióban van, A példa szerint a három gép alapbeállítása csupán annyiban különbözött egymástól, hogy a második keverőtárcsa-beállítása az elsőhöz képest egy hellyel előre, S+1 állásba volt forgatva, a harmadik pedig az elsőhöz képest hárommal, S+3 pozícióba.

A megfelelő napi beállítás megtalálásához próbálgatni kell a beállításokat. Ahogyan az első gépben megcserélik a keverőtárcsák sorrendjét, ugyanúgy meg kell cserélni a másik kettőben is, az első keverőtárcsa beállításához képest a másodikat egy betűhellyel, a harmadikat hárommal előre kell forgatni.

Turing elektromos vezetékkel összekötötte a felhasznált gépek bemenetét és kimenetét a betűláncnak megfelelően (42.ábra), aminek eredményeként a lánc által képzett hurok párhuzamossá vált az elektromos áramkör hurkával. Elgondolása szerint az áram csak akkor halad végig mindhárom gépen, ha mindhárom beállítása helyes. Turing kiküszöbölte a kapcsolótábla hatását, következésképpen annak sokmilliárdnyi beállításával nem kellett számolnia.

Ezután csak végig kellett pörgetni a maximum 1.054.560 beállítást, minden Enigma beállítását egyszerre módosítva.

Amikor zárt az áramkör, akkor volt jó pozícióban a keverőtárca, tehát megtaláltuk az aznapi beállítást.

Ennek automatizálásához készítette el a berendezését.

A német haditengerészetnél az Enigma egy bonyolultabb változatát használták. Nem öt, hanem nyolc keverőtárcsa közül lehetett választani.

Ráadásul volt még egy különbség, a visszairányítót huszonhatféle helyzetben lehetett rögzíteni (forgatni), így a lehetséges kulcsok száma még a huszonhatszorosára emelkedett.

Purple

A Purple (Bíbor) a japán rejtjelező gép, alapjaiban hasonlít az Enigmára (43. ábra).
Négy tárcsát használ
1942-ben feltörték az amerikaiak

43. ábra Purple

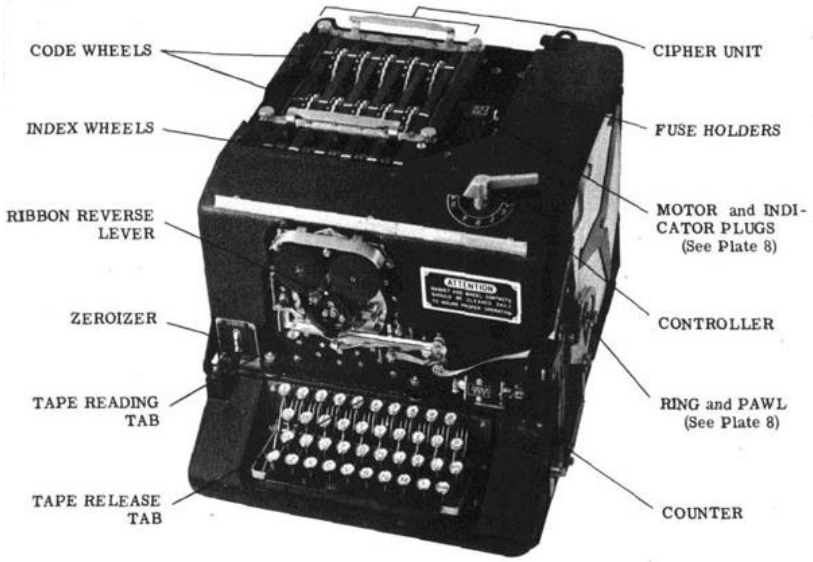
Typex MK-III

Bonyolultabb mint az Enigma, ezért nem tudták feltörni. Öt tárcsát használ (44. ábra), így a kiinduló pozíciók száma csak a tárcsákra vonatkoztatva: 26x26x26x26x26 = 11.881.376

44. ábra Types MK-III (saját kép)

SIGABA

1941-től kezdve használja US NAVY. Öt tárcsás (45. ábra), mert képesek törni a négy tárcsás Purple-t, így ők egy nagyságrenddel bonyolultabbat használnak, hogy biztonságban tudják az üzenetváltásaikat. Ez a szemlélet még vissza fog köszönni a II. Világháború után és tart a mai napig, ahogy majd a következő részből megtudhatjuk.



45. ábra Types MK-III (saját kép)

IV. Irodalomjegyzék

[1] Simon Singh: Kódkönyv - A rejtjelezés és rejtjelfejtés története, 2007, ISBN: 9789635307982
[2] Virasztó Tamás: Titkosítás és adatrejtés, 2004, ISBN: 9789632142531.
[3] David Kahn: The Codebreakers- The Comprehensive History of Secret Communication from Ancient Times to the Internet, 1997, ISBN: 9780684831305
[4] Megyesi Zoltán: Titkosírások, 1999, ISBN: 9639178853

V. mellékletek

1. melléklet

Caesar-kódolás Python forráskódja

```
#Caesar-kódolás
#A=65, Z=90

def kodolas():
    print("Caesar-kódolású üzenet előállítása")
    uzenet=input("Mi legyen a titkosítandó üzenet? ")
    nagybetus_uzenet=uzenet.upper()
    print(nagybetus_uzenet)
    eltolas=int(input("Mekkora eltolással titkosítsam? (1-25)"))
    titkositott_uzenet=""
    if eltolas<26 and eltolas>0:
        for i in range(len(nagybetus_uzenet)):
            if ord(nagybetus_uzenet[i])+eltolas>90:
                ujbetu=chr(ord(nagybetus_uzenet[i])-26+eltolas)
            else:
                ujbetu=chr(ord(nagybetus_uzenet[i])+eltolas)
            titkositott_uzenet=titkositott_uzenet+ujbetu
        print(titkositott_uzenet)

def dekodolas():
    print("Caesar-kódolású üzenet visszafejtése")
    visszafejtendo_uzenet=input("Mi legyen a visszafejtendő üzenet? ")
    nagybetus_uzenet=visszafejtendo_uzenet.upper()
    print(nagybetus_uzenet)
    eltolas=int(input("Mekkora eltolással titkosították? (1-25)"))
    visszafejtett_uzenet=""
    if eltolas<26 and eltolas>0:
        for i in range(len(nagybetus_uzenet)):
            if ord(nagybetus_uzenet[i])-eltolas<65:
                ujbetu=chr(ord(nagybetus_uzenet[i])+26-eltolas)
            else:
                ujbetu=chr(ord(nagybetus_uzenet[i])-eltolas)
            visszafejtett_uzenet=visszafejtett_uzenet+ujbetu
        print(visszafejtett_uzenet)

#főprogram
print("Caesar-kódolás bemutatása")
menu=int(input("Titkosítást(1), vagy visszafejtést(2) végezzek?"))
if menu==1:
    kodolas()
if menu==2:
    dekodolas()
```

2. melléklet

Caesar-kódolás törését is tartalmazó Python forráskód

```
#Caesar-kódolás
#A=65, Z=90
def kodolas():
    print("Caesar-kódolású üzenet előállítása")
    uzenet=input("Mi legyen a titkosítandó üzenet? ")
    nagybetus_uzenet=uzenet.upper()
    print(nagybetus_uzenet)
    eltolas=int(input("Mekkora eltolással titkosítsam? (1-25)"))
    titkositott_uzenet=""
    if eltolas<26 and eltolas>0:
        for i in range(len(nagybetus_uzenet)):
            if ord(nagybetus_uzenet[i])+eltolas>90:
                ujbetu=chr(ord(nagybetus_uzenet[i])-26+eltolas)
            else:
                ujbetu=chr(ord(nagybetus_uzenet[i])+eltolas)
            titkositott_uzenet=titkositott_uzenet+ujbetu
        print(titkositott_uzenet)
def dekodolas():
    print("Caesar-kódolású üzenet visszafejtése")
    visszafejtendo_uzenet=input("Mi legyen a visszafejtendő üzenet? ")
    nagybetus_uzenet=visszafejtendo_uzenet.upper()
    print(nagybetus_uzenet)
    eltolas=int(input("Mekkora eltolással titkosították? (1-25)"))
    visszafejtett_uzenet=""
    if eltolas<26 and eltolas>0:
        for i in range(len(nagybetus_uzenet)):
            if ord(nagybetus_uzenet[i])-eltolas<65:
                ujbetu=chr(ord(nagybetus_uzenet[i])+26-eltolas)
            else:
                ujbetu=chr(ord(nagybetus_uzenet[i])-eltolas)
            visszafejtett_uzenet=visszafejtett_uzenet+ujbetu
        print(visszafejtett_uzenet)

def tores():
    print("Caesar-kódolású üzenet törése (minden eltolás elvégzése)")
    torendo_uzenet=input("Mi legyen a törendő üzenet? ")
    nagybetus_uzenet=torendo_uzenet.upper()
    print(nagybetus_uzenet)
    for eltolas in range(26):
        visszafejtett_uzenet=""
        for i in range(len(nagybetus_uzenet)):
            if ord(nagybetus_uzenet[i])-eltolas<65:
                ujbetu=chr(ord(nagybetus_uzenet[i])+26-eltolas)
            else:
                ujbetu=chr(ord(nagybetus_uzenet[i])-eltolas)
            visszafejtett_uzenet=visszafejtett_uzenet+ujbetu
        print(visszafejtett_uzenet)
#főprogram
print("Caesar-kódolás bemutatása")
menu=int(input("Titkosítást(1), visszafejtést(2), vagy törést(3)
végezzek?"))
```

```
if menu==1:
    kodolas()
if menu==2:
    dekodolas()
if menu==3:
    tores()
```

3. melléklet

A Vigenére módszerrel történő titkosítás és dekódolás Python forráskódja

```
   import time
   def kodolas(uzenet, kulcs):
       kulcs_hossza = len(kulcs)
       uzenet_ascii_kod_listaja = [ord(i) for i in uzenet]
       print(uzenet_ascii_kod_listaja)
       kulcs_ascii_kod_listaja = [ord(i) for i in kulcs]
       print(kulcs_ascii_kod_listaja)
       titkositott_uzenet = ''
       print("A titkosított üzenet:")
       for i in range(len(uzenet_ascii_kod_listaja)):
           abc_hanyadik_karaktere = (uzenet_ascii_kod_listaja[i] +
kulcs_ascii_kod_listaja[i % kulcs_hossza]) % 26
           titkositott_uzenet += chr(abc_hanyadik_karaktere + 65)
           time.sleep(1)
           print(chr(abc_hanyadik_karaktere + 65),end='')


   def dekodolas(titkositott_uzenet, kulcs):
       kulcs_hossza = len(kulcs)
       titkositott_uzenet_ascii_kod_listaja = [ord(i) for i in titkositott_uzenet]
       print(titkositott_uzenet_ascii_kod_listaja)
       kulcs_ascii_kod_listaja = [ord(i) for i in kulcs]
       print(kulcs_ascii_kod_listaja)
       eredeti_uzenet = ''
       print("Az eredeti üzenet:")
       for i in range(len(titkositott_uzenet_ascii_kod_listaja)):
           abc_hanyadik_karaktere = (titkositott_uzenet_ascii_kod_listaja[i] -
kulcs_ascii_kod_listaja[i % kulcs_hossza]) % 26
           eredeti_uzenet += chr(abc_hanyadik_karaktere + 65)
           time.sleep(1)
           print(chr(abc_hanyadik_karaktere + 65), end='')

   #főprogram
   print("Vigenére-kódolás bemutatása")
   menu=int(input("Titkosítást(1), visszafejtést(2) végezzek? "))
   if menu==1:
       titkositando_uzenet=input("Mi a titkosítandó üzenet? ")
       nagybetus_uzenet=titkositando_uzenet.upper()
       titkosito_kulcs = input("Mi legyen a titkosításhoz használt kulcsszó? ")
       nagybetus_kulcs = titkosito_kulcs.upper()
       print(nagybetus_kulcs)
       kodolas(nagybetus_uzenet,nagybetus_kulcs)
   if menu==2:
       visszafejtendo_uzenet=input("Mi a titkosított üzenet? ")
       nagybetus_uzenet=visszafejtendo_uzenet.upper()
       titkosito_kulcs = input("Mi volt a titkosításhoz használt kulcsszó? ")
       nagybetus_kulcs = titkosito_kulcs.upper()
       print(nagybetus_kulcs)
       dekodolas(nagybetus_uzenet, nagybetus_kulcs)
```

Univerzita J. Selyeho
Fakulta ekonómie a informatiky
Hradná ul. 21.
SK-945 01 Komárno
fei.ujs.sk

Dr. Kiss Gábor

Kriptográfiai alapismeretek I.
Ókortól a II. Világháború végéig

Recenzenti/Recenzensek:
Ing. Ladislav Tóth, PhD. (Slovenská Poľnohospodárska Univerzita, Technická fakulta)
RNDr. Zuzana Árki, PhD. (Selye János Egyetem, Matematika Tanszék)

Vydavateľ - Kiadó:
Univerzita J. Selyeho - Selye János Egyetem

Rozsah/Terjedelem:
5 AH / 5 szerzői ív

Rok vydania/Megjelenés éve
2025

Číslo vydania/Kiadás száma:
Prvé vydanie/Első kiadás

ISBN 978-80-8122-525-3
EAN 9788081225253