Правительство Российской Федерации
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ
«ВЫСШАЯ ШКОЛА ЭКОНОМИКИ»
(НИУ ВШЭ)

Московский институт электроники и математики им. А.Н. Тихонова

ОТЧЕТ
О ПРАКТИЧЕСКОЙ РАБОТЕ № 3
по дисциплине «Математические основы защиты информации»
Шифры гаммирования

Студент гр. БИБ201
Д.А. Морин
«23» 04.2022 г.

Руководитель
Заведующий кафедрой
информационной
безопасности киберфизических систем
канд. техн. наук, доцент
О.О. Евсютин
«___» _____ 2022 г.

Москва 2022

Содержание

1. Краткие теоретические сведения

Шифр - система обратимых преобразований, зависящая от некоторого секретного параметра (ключа) и предназначенная для обеспечения секретности передаваемой информации.

Ключ - это часть информации, обычно представляющая собой строку цифр или букв, хранящуюся в файле, которая при обработке с помощью криптографического алгоритма может кодировать или декодировать криптографические данные.

Шифрование (зашифрование) - обратимое преобразование информации в целях сокрытия от неавторизованных лиц с предоставлением в это же время авторизованным пользователям доступа к ней.

Расшифрование - процесс преобразования зашифрованных данных в открытые данные при помощи шифра.

Дешифрование (дешифровка) - процесс извлечения открытого текста без знания криптографического ключа на основе известного шифрованного.

Открытый текст - в криптографии исходный текст, подлежащий шифрованию, либо получившийся в результате расшифровки. Может быть прочитан без дополнительной обработки (без расшифровки).

Шифротекст, шифртекст, криптограмма - результат операции шифрования.

Частотный анализ, частотный криптоанализ - один из методов криптоанализа, основывающийся на предположении о существовании нетривиального статистического распределения отдельных символов и их последовательностей, как в открытом тексте, так и в шифротексте, которое, с точностью до замены символов, будет сохраняться в процессе шифрования и дешифрования. Частотный анализ предполагает, что частотность появления заданной буквы алфавита в достаточно длинных текстах одна и та же для разных текстов одного языка.

Полный перебор предполагает перебор всех возможных вариантов ключа шифрования до нахождения искомого ключа.

Шифр Виженера — метод полиалфавитного шифрования буквенного текста с использованием ключевого слова. В шифре Виженера в качестве ключа шифрования обычно используется короткая фраза, называемая лозунгом (паролем), которая циклически повторяется.

Самоключ Виженера - в качестве начального ключа брать только один символ, к которому добавляются все символы открытого текста, за исключением последнего (самоключ Виженера по открытому тексту) или добавлять к начальному символу поочередно символы шифртекста (самоключ Виженера по шифртексту).

Шифр Цезаря — это вид шифра подстановки, в котором каждый символ в открытом тексте заменяется символом находящимся на некотором постоянном числе позиций левее или правее него в алфавите.

2. “Ручное” шифрование и расшифрование

Шифр Виженера

1) Повторением короткого лозунга

Зашифруем слово moment = (12, 14, 12, 4, 13, 19) с помощью ключа key (10, 4, 24) (таблица 1). Если длины ключа не хватает на все слово, то оно дублируется (как в примере ниже).

Таблица 1

text	m	o	m	e	n	t
num text	12	14	12	4	13	19
num key	10	4	24	10	4	24
sum nums	22	18	10	14	17	17
sum word	w	s	k	o	r	r

Итак, мы получили зашифрованное слово wskorr.

Расшифруем полученный шифртекст с помощью нашего ключа (таблица 2).

Таблица 2

text	w	s	k	o	r	r
num text	22	18	10	14	17	17
num key	10	4	24	10	4	24
dif nums	12	14	12	4	13	19
dif word	m	o	m	e	n	t

Получили открытый текст, который и передавали при шифровании.

В качестве другого примера возьмем слово embassy (4, 12, 1, 0, 18, 18, 24) и ключ peace (15, 4, 0, 2, 4) (таблица 3)

Таблица 3

text	e	m	b	a	s	s	y
num sum	4	12	1	0	18	18	24

num key	15	4	0	2	4	15	4
sum nums	19	16	1	2	22	7	2
sum word	t	q	b	c	w	h	c

tqbcwhc - зашифрованное слово. Теперь расшифруем его (таблица 4)

Таблица 4

text	t	q	b	c	w	h
num text	19	16	1	2	22	7
num key	15	4	0	2	4	15
dif nums	4	12	1	0	18	18
dif word	e	m	b	a	s	s

Получили знакомый открытый текст embassy.

2) Самоключ Виженера по открытому тексту.

Зашифруем слово moment = (12, 14, 12, 4, 13, 19) с помощью самоключа с добавлением буквы k (10) (таблица 5). Ключ получается путем присоединения к началу слова буквы k и удалением последней буквы t.

Таблица 5

text	m	o	m	e	n	t
num text	12	14	12	4	13	19
num key (num text)	10	12	14	12	4	13
sum nums	22	0	0	16	17	6
sum word	w	a	a	q	r	g

Итак, мы получили зашифрованное слово waaqrg.

Расшифруем полученный шифртекст с помощью нашего ключа (таблица 6).

Таблица 6

text	w	a	a	q	r	g
num text	22	0	0	16	17	6
num key	10	12	14	12	4	13
dif nums	12	14	12	4	13	19
dif word	m	o	m	e	n	t

Получили открытый текст, который и передавали при шифровании.

В качестве другого примера возьмем слово embassy (4, 12, 1, 0, 18, 18, 24) и ключ p (15) (таблица 7)

Таблица 7

text	e	m	b	a	s	s	y
num sum	4	12	1	0	18	18	24
num key (num text)	15	4	12	1	0	18	18
sum nums	19	16	13	1	18	10	16
sum word	t	q	n	b	s	k	q

tqnbskq - зашифрованное слово. Теперь расшифруем его (таблица 8)

Таблица 8

text	t	q	n	b	s	k	q
num text	19	16	13	1	18	10	16
num key	15	4	12	1	0	18	18
dif nums	4	12	1	0	18	18	24
dif word	e	m	b	a	s	s	y

Получили знакомый открытый текст embassy.

3) Самоключ Виженера по шифртексту.

Зашифруем слово moment = (12, 14, 12, 4, 13, 19) с помощью самоключа с добавлением буквы k (10) (таблица 9). Ключ есть шифртекст, у которого впереди добавлена буква k и без последней буквы g.

Таблица 9

text	m	o	m	e	n	t
num text	12	14	12	4	13	19
num key (sum nums)	10	22	10	22	0	13
sum nums	22	10	22	0	13	6
sum word	w	k	w	a	n	g

Итак, мы получили зашифрованное слово wkwang.

Расшифруем полученный шифртекст с помощью нашего ключа (таблица 10).

Таблица 10

text	w	k	w	a	n	g
num text	22	10	22	0	13	6
num key	10	22	10	22	0	13
dif nums	12	14	12	4	13	19
dif word	m	o	m	e	n	t

Получили открытый текст, который и передавали при шифровании.

В качестве другого примера возьмем слово embassy (4, 12, 1, 0, 18, 18, 24) и ключ p (15) (таблица 11)

Таблица 11

text	e	m	b	a	s	s	y
num sum	4	12	1	0	18	18	24
num key	15	19	5	6	6	24	16

(sum nums)							
sum nums	19	5	6	6	24	16	14
sum word	t	f	g	g	y	q	o

tfggyqo - зашифрованное слово. Теперь расшифруем его (таблица 12)

Таблица 12

text	t	f	g	g	y	q	o
num text	19	5	6	6	24	16	14
num key	15	19	5	6	6	24	16
dif nums	4	12	1	0	18	18	24
dif word	e	m	b	a	s	s	y

Получили знакомый открытый текст embassy.

3. Программная реализация зашифрования и расшифрования

Шифр Виженера

1) Повторением короткого лозунга

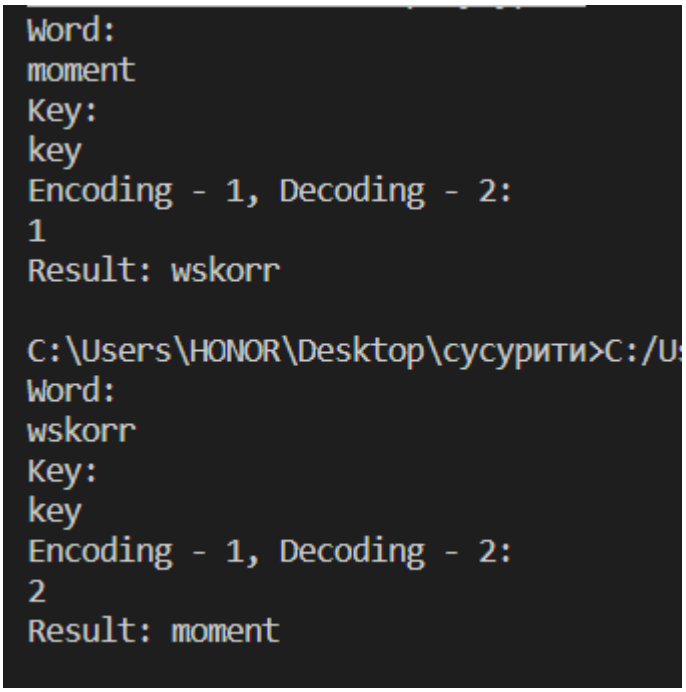

Рисунок 1 - результат зашифрования и расшифрования с помощью программы 1.py

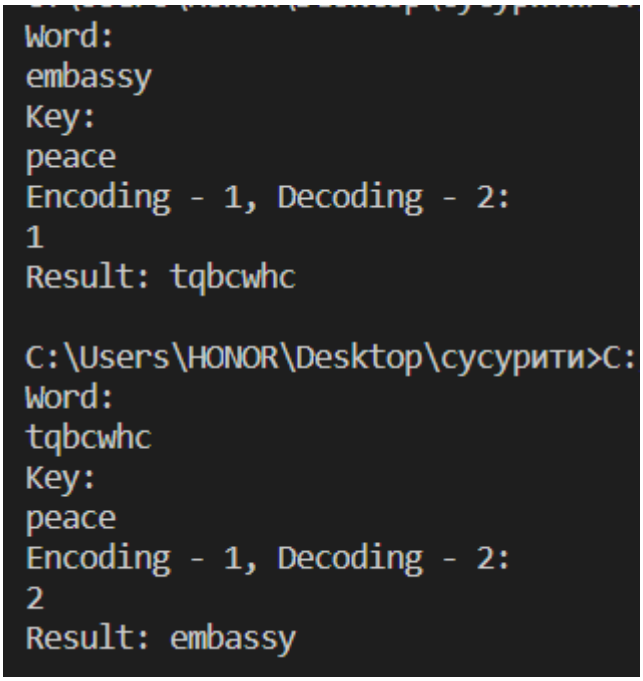

Рисунок 2 - результат зашифрования и расшифрования с помощью программы 1.py

2) Самоключ Виженера по открытому тексту.

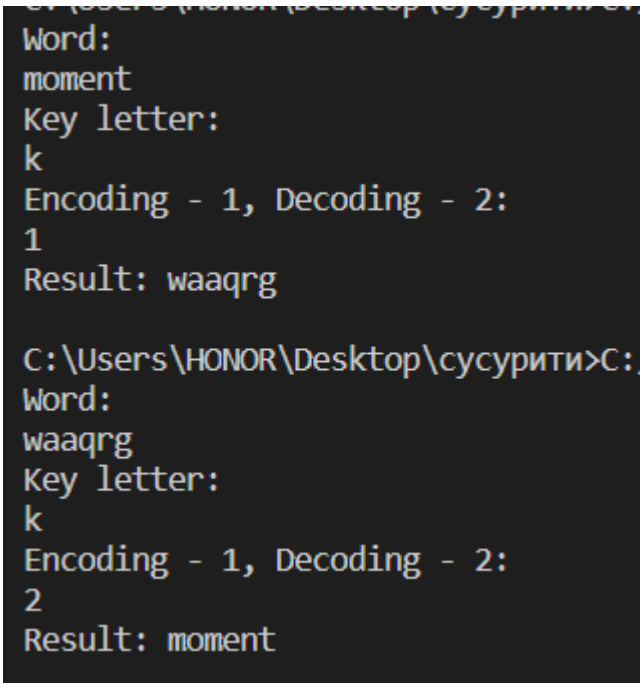

Рисунок 3 - результат зашифрования и расшифрования с помощью программы 2.py

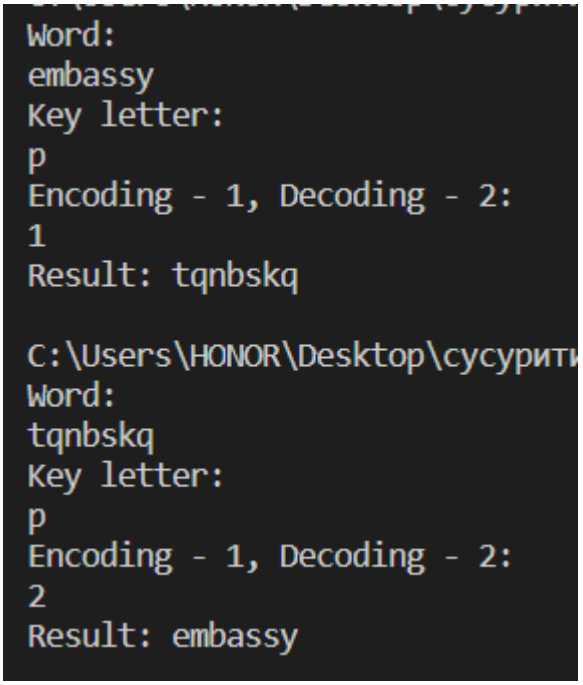

Рисунок 4 - результат зашифрования и расшифрования с помощью программы 2.py

3) Самоключ Виженера по шифртексту.

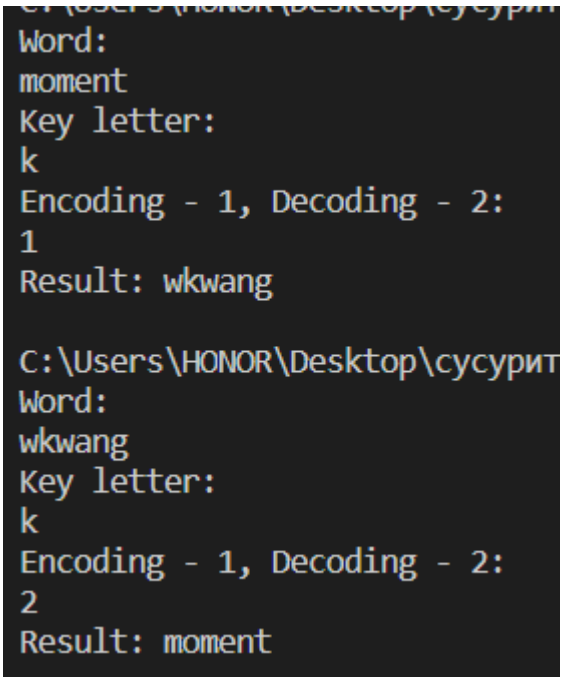

Рисунок 5 - результат зашифрования и расшифрования с помощью программы 3.py

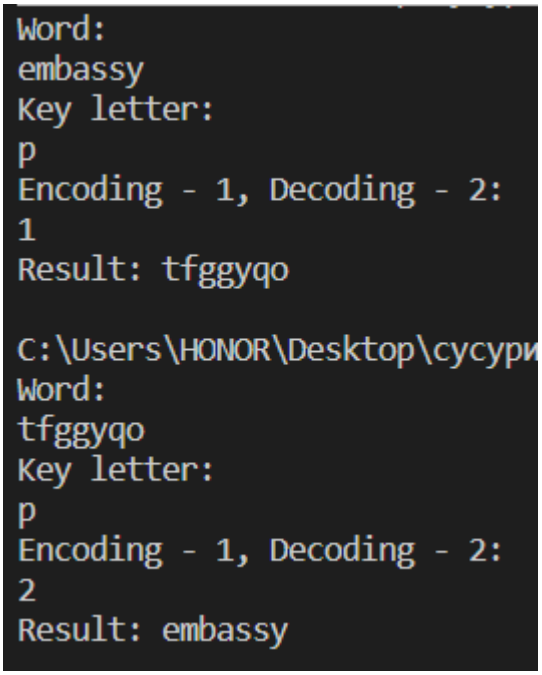

Рисунок 6 - результат зашифрования и расшифрования с помощью программы 3.py

4. Криптоанализ шифров

Пусть p — длина ключевого слова. Обычно криптоанализ шифра Виженера проводится в два этапа. На первом этапе определяется число p, на втором этапе — само ключевое слово.

Основной слабостью шифра Виженера является повторяющийся характер его ключа. Если криптоаналитик правильно угадывает длину ключа, то текст шифра можно рассматривать как переплетенные шифры Цезаря, которые по отдельности легко ломаются. Тесты Касиски и Фридмана могут помочь определить длину ключа.

Экзамен Касиски, также называемый тестом Касиски, использует тот факт, что повторяющиеся слова могут случайно иногда шифроваться с использованием одних и тех же ключевых букв, что приводит к повторным группам в зашифрованном тексте. Например:
Ключ: ABCDABCDABCDABCDABCDABCDABCD
Открытый текст: CRYPTO ISSHORTFOR CRYPTO GRAPHY
Шифротекст: CSASTP KVSIQUTGQU CSASTP IUAQJB

В зашифрованном тексте можно увидеть повторение CSASTP, расстояние между которыми равно 16. Предполагая, что сегменты представляют одни и те же сегменты открытого текста, это означает, что длина ключа 1, 2, 4, 8, 16 символов. Так как длина ключа 1 или 2 очень короткая, нужно попробовать только 4, 8, 16. Чем длиннее сообщение, тем более точный тест, потому что они обычно содержат больше повторяющихся сегментов шифртекста.

Другой пример:
Зашифрованный текст: VHVS SP QUCE MRVBNBBB VHVS URQGIBDUGRNICJ QUCE RVUAXSSR

Расстояние между повторениями VHVS - 18. Можно предположить, что длина ключа 1, 2, 3, 6, 9, 18 символов. Расстояние между повторениями QUCE составляет 30 символов => длина ключа может быть 1, 2, 3, 5, 6, 10, 15, 30 символов. Взяв пересечение этих множеств, можно с уверенностью заключить, что наиболее вероятная длина ключа равна 6, поскольку 3, 2 и 1 нереально коротки.

Тест Фридмана

Фридман использовал индекс совпадения, который измеряет неравномерность частот букв шифра, чтобы взломать шифр. Зная вероятность k(p) того, что любые две случайно выбранные буквы исходного языка совпадают и вероятность совпадения для равномерного

случайного выбора из алфавита k(r) (1/26 = 0,0385 для английского), длину ключа можно оценить следующим образом (рисунок 7).

$$\frac{\kappa_p - \kappa_r}{\kappa_o - \kappa_r}$$

Рисунок 7 - оценка длины ключа

Где наблюдаемая частота совпадений (рисунок 8):

$$\kappa_o = \frac{\sum_{i=1}^{c} n_i(n_i - 1)}{N(N-1)}$$

Рисунок 8 - наблюдаемая частота совпадений букв

Где c - размер алфавита (26 для английского языка), N - длина текста, а n_i при i от 1 до n - наблюдаемые частоты букв зашифрованного текста в виде целых чисел.

Однако это только приближение, точность которого увеличивается с размером текста. Лучший подход для шифров с повторяющимся ключом состоит в том, чтобы скопировать зашифрованный текст в строки матрицы, имеющие столько столбцов, сколько предполагаемая длина ключа, а затем вычислить средний индекс совпадения с каждым столбцом, рассматриваемым отдельно; когда это делается для каждой возможной длины ключа, самый высокий средний индекс совпадения соответствует наиболее вероятной длине ключа. Такие тесты могут быть дополнены информацией из экзамена Касиски.

Разберем полный пример на основе русского алфавита с шифротекстом (и с помощью программы maxsubstr.py найдём наибольшую подстроку с большим числом повторения):

glypwjzsjgwbuzripawjldyqarkkoamnzurpqmzsiqdijaqfpwyleqrwsslpstkxsjkuplc
euwgyzsxfwqirxckozfqqrsipvucelzckkovuawiglmrocwfyhcpvrwpiulsfsvisjrvucx
yjrsxdagbwurowlpfvdvpwohrqnjmnkzuopcogwlnijehpxcmlchylcwrwbudbhakag
xdipooegyxfafuhfmaagtdxpgjyzqkagnschcwjkqrokvcwbvwgspgkcdxafeqylcelqa
shbsdycdsexcngfumsklikhbpggsuhfmaagkkkxawbkdvorksrfrsrdsixcmlcvzjrjpael
hxgwnoulyayrsjlxwraourpgmjbvfdmlcqreviqpvvzvelyoelxxsnbshrsmgsulxxmpv
vlxxcnbvwewswzcbgmrdhyhkmbktrfkfjaprvsgyhzpwrilswwlswycsehbmajodhps
ppvvpkmliskkyhzukylmlyszrqswqahlslyrpvvwovkswwlkwuazcdcxfahvfrrmhcx
bsxqazwkkwcrccyohoqwkhkfgpgzqmigpkrvpmpohtrsrczwedlsspoegswlkklvohk
kfvebsyzzpskvreqlokvjuppwrieabvukpnqpclmxmisrqkagzsidxkcktnlbijagjfyqkq
bzfkxgkbkhmllkzfjsiqswwlewcodvurenohyhwsqpjzvsfjaarqsjcohrwsslktnlpmgo

hyhmsdbsvvrsnhogwytrqbvgmspzzvvcpwebkrkajwbrqnlcjqvlxxmpvvzyvjzkzgo
acxplwcskadyrxisosivwmedhrocszarflxkgpppzsjgrczsfsgysfyovrdszqdipjsksbsrk
qfozlmjsjhxezhslvovqpcjsoeipcfwripojzddlcebkhbrcphyhsransrvsrewjrlvezezzw
isdswwlwiyjgkkkxnacgooagpvmrstndcehcgyjijhdlciafuoelzafuopggsdrlmjadyrxi
qpocnsreswkkpvgabuvkrbyccooeeqsjrfippvvlxxcnbvwpvmihyhcekaqfipicovfssr
udwtkdlcuqfqxiaphyhsvjwdkrzwrkhyhgsphrnlniuapnlpmgoijhnmlioebyxfafrszp
gyoklyrqwgnhvpqkavwopcrwjlyrqwfvjymlckzisejhcnlxktesnhbwrkkrqnipwpfxd
xfawikyyqagnldlrdszuyallciwkfjagtuoilocehmskloebbiaabkoisdbsihneawavukxf
whtrxrcyhjwyxfawewovlahmlkagbwrovsuebxsosnhskroqyezgkyxmohfibmcjrjd
xhakzchkksagulbiapzpibskpvvlbgyisidcekkfvperbwbveexuervvzvcwrlvosdswwl
swgjqfpwyleq

В данном тексте обнаружено четырехкратное повторение буквосочетания «swwl». Расстояние между ними: 63 = 7*3^2, 182 = 2*7*13, 266 = 2*7*19, 483 = 3*7*23. НОД = 7 - предполагаемая длина ключа.

Разбиваем шифртекст по 7 символов и записываем в матрицу с семью столбцами (воспользовался программой count.py):

glypwjz

sjgwbuz

ripawjl

dyqarkk

oamnzur

pqmzsiq

dijaqfp

wyleqrw

sslpstk

xsjkupl

ceuwgyz

sxfwqir

xckozfq

qrsipvu

celzckk

ovuawig

lmrocwf

yhcpvrw

piulsfs

visjrvu

cxyjrsx

dagbwur

owlpfvd

vpwohrq

njmnkzu

opcogwl

nijehpx

cmlchyl

cwrwbud

bhakagx

dipooeg

yxfafuh

fmaagtd

xpgjyzq

kagnsch

cwjkqro

kvcwbvw

gspgkcd

xafeqyl

celqash

bsdycds

excngfu

msklikh

bpggsuh

fmaagkk

kxawbkd

vorksrf

rsrdsix

cmlcvzj

rjpaelh

xgwnoul

yayrsjl

xwraour

pgmjbvf

dmlcqre

viqpvvz

velyoel

xxsnbsh

rsmgsul

xxmpvvl

xxcnbvw

ewswzcb

gmrdhyh

kmbktrf

kfjaprv

sgyhzpw

rilswwl

swycseh

bmajodh

psppvvp

kmliskk

yhzukyl

mlyszrq

swqahls

lyrpvvw

ovkswwl

kwuazcd

cxfahvf

rrmhcxb

sxqazwk

kwcrccy

ohoqwkh

kfgpgzq

migpkrv

pmpohtr

srczwed

lsspoeg

swlkklv

ohkkfve

bsyzzps

kvreqlo

kvjuppw

rieabvu

kpnqpcl

mxmisrq

kagzsid

xkcktnl

bijagjf

yqkqbzf

kxgkbkh

mllkzfj

siqswwl

ewcodvu

renohyh

wsqpjzv

sfjaarq

sjcohrw

sslktnl

pmgohyh

msdbsvv

rsnhogw

ytrqbvg

mspzzvv

cpwebkr

kajwbrq

nlcjqvl

xxmpvvz

yvjzkzg

oacxplw

cskadyr

xisosiv

wmedhro

cszarfl

xkgpppz

sjgrczs

fsgysfy

ovrdszq

dipjsks

bsrkqfo

zlmjsjh

xezhslv

ovqpcjs

oeipcfw

ripojzd

dlcebkh

brcphyh

sransrv

srewjrl

vezezzw

isdswwl

wiyjgkk

kxnacgo

oagpvmr

stndceh

cgyjijh

dlciafu

oelzafu

opggsdr

lmjadyr

xiqpocn

sreswkk

pvgabuv

krbycco

oeeqsjr

fippvvl

xxcnbvw

pvmihyh

cekaqfi

picovfs

srudwtk

dlcuqfq

xiaphyh

svjwdkr

zwrkhyh

gsphrnl

niuapnl

pmgoijh

nmlioeb

yxfafrs

zpgyokl

yrqwgnh

vpqkavw

opcrwjl

yrqwfvj

ymlckzi

sejhcnl

xktesnh

bwrkkrq

nipwpfx

dxfawik

yyqagnl

dlrdszu

yallciw

kfjagtu

oiloceh

mskloeb

biaabko

isdbsih

neawavu

kxfwhtr

xrcyhjw

yxfawew

ovlahml

kagbwro

vsuebxs

osnhskr

oqyezgk

yxmohfi

bmcjrjd

xhakzch

kksagul

biapzpi

bskpvvl

bgyisid

cekkfvp

erbwbve

exuervv

zvcwrlv

osdswwl

swgjqfp

wyleq

Частота повторения букв в столбцах (посчитал опять же с помощью программы count.py):

1 столбец:

a	b	c	d	e	f	g	h	i	j	k	l	m
0	14	15	11	5	4	4	0	2	0	21	4	8

n	o	p	q	r	s	t	u	v	w	x	y	z
7	21	10	1	10	22	0	0	8	5	20	15	4

2 столбец:

a	b	c	d	e	f	g	h	i	j	k	l	m
16	0	4	0	14	4	5	6	15	10	4	11	18

n	o	p	q	r	s	t	u	v	w	x	y	z
0	5	10	10	13	20	9	0	8	14	10	5	0

3 столбец:

a	b	c	d	e	f	g	h	i	j	k	l	m
13	1	22	5	5	8	20	0	1	14	10	21	12

n	o	p	q	r	s	t	u	v	w	x	y	z
6	1	9	15	14	7	1	8	0	3	0	8	7

4 столбец:

a	b	c	d	e	f	g	h	i	j	k	l	m
32	4	5	7	12	0	4	7	7	12	17	4	0

n	o	p	q	r	s	t	u	v	w	x	y	z
9	16	23	6	4	7	0	3	0	17	1	6	8

5 столбец:

a	b	c	d	e	f	g	h	i	j	k	l	m
15	15	11	4	1	6	12	19	3	3	8	0	0

n	o	p	q	r	s	t	u	v	w	x	y	z
0	11	8	13	8	28	3	1	11	17	0	1	14

6 столбец:

a	b	c	d	e	f	g	h	i	j	k	l	m
1	0	9	3	10	12	4	0	10	12	17	7	2

n	o	p	q	r	s	t	u	v	w	x	y	z
8	0	7	2	20	3	10	16	21	8	2	13	13

7 столбец:

a	b	c	d	e	f	g	h	i	j	k	l	m
0	4	0	11	3	9	8	28	4	3	11	32	0

n	o	p	q	r	s	t	u	v	w	x	y	z
1	8	4	15	14	17	0	11	12	12	5	2	0

Воспользуемся формулой на рисунке 9 для нахождения сдвига столбцов относительно первого.

$$MI(\vec{x},\vec{y}) = \sum_{i=1}^{m} \frac{f_i g_i}{nn'}$$

Рисунок 9 - формула взаимного индекса совпадения, где f_i, g_i это число i-ого символа алфавита в двух строках (x и y), а n, n' это длины строк.

С помощью программы math.py посчитал числитель вышеприведенной формулы, для ответа нужно лишь разделить на длину строк, которую можно легко узнать через len(string) (с помощью языка python). Получилось, что длины с 1 по 5 строки = 211, а 6 и 7 = 210

для 2 строки: MI = (1662/211)/211 = 0.0373 => сдвиг равен 6 или 10
для 3 строки: MI = (1701/211)/211 = 0.0382 => сдвиг равен 12, 10 или 7
для 4 строки: MI = (1727/211)/211 = 0.0387 => сдвиг равен 12, 10 или 7
для 5 строки: MI = (1934/211)/211 = 0.0434 => сдвиг равен 4 или 11
для 6 строки: MI = (1719/211)/210 = 0.0388 => сдвиг равен 7 или 12
для 7 строки: MI = (1722/211)/210 = 0.0389 => сдвиг равен 7 или 12

Составим уравнения для выявления ключевого слова:
g1 - g2 = |6| g1 - g3 = |12| g1 - g4 = |12|
g1 - g5 = |4| g1 - g6 = |7| g1 - g7 = |7|

Осталось найти значение g1- перебираем его от 1 до 26:
auomeht
bvpnfiu
cwqogjv
dxrphkw
eysqilx
fztrjmy
gausknz
hbvtloa
icwumpb
jdxvnqc
keyword

lfzxpse
mgayqtf
nhbzrug
oicasvh
pjdbtwi
qkecuxj
rlfdvyk
smgewzl
tnhfxam
uoigybn
vpjhzco
wqkiadp
xrljbeq
ysmkcfr
ztnldgs

Таким образом, получили ключевое слово keyword, расшифруем текст с помощью программы 1.py:

whatiswifiandwhereisitusedtheworldofmoderntelecommunicationtechnologyisa
washwithacronymslongnumbersandotherweirdbitsofcodethatfewpeopleundersta
ndbutwifidoesntreallystandforwirelessfidelityusingthisstandardcomputersandoth
erdevicescanlinkinawirelesslocalareanetworkwlanwhichisanumberofcomputerso
rcomputerlikedevicesthatcantalktoeachotherusinghighfrequencyradiowavesinste
adofconnectingcablesthewlancaninturnbehookedintotheinternetusuallywiththeai
dofacablebasicallythenwifiisagenericnameforthemainmethodbywhichawlanisset
upbutthetermwifiaswellasthetechnologyitselfhasevolvedquiteabitsinceitwasfirstc
oinedinaboutandisnowusedmorebroadlyparticularlybythegeneralpublictomeana
widerangeofwirelesscommunicationtechnologieswifiusesperhapsthemostvisible
manifestationofwifiisthecoffeeshoplaptoptunedcordlesslyintoawlanandhenceinto
theworldwidewebbutsomephoneusersmightalsobedoingitbywifivoipvoiceoverth
einternetprotocolphonesenableuserstospeaktoothersviatheinternettheincreasinga
vailabilityofwifimeansthatpeoplewithvoipphonescanusethemmoreandmorelikem
obilephonestalkingwithfriendsandcolleaguesovertheinternetfromthesamecoffees
hopinwhichtheyconnecttheirlaptopstotheworldwidewebwifiisusedinmanyothera
pplicationsaswellsometelevisionsaregoingwifiallowingviewerstowanderaboutthe
irhouseswiththeirownportablescreensonecompanyrecentlyofferedacamerathatco

nnectstotheinternetviawifiallowingpeopletoemailphotostofriendsandcolleaguesdirectlyfromtheircamerasamoremundanebutwidespreaduseofwifiisincommunic

Что касается самоключа Виженера, то мы знаем, что длина ключа это сам текст (открытый или зашифрованный) с добавлением одной буквы => можно перебрать все буквы и найти такой ключ, при котором открытый текст будет осмысленным

5. Вывод

По зашифрованию и расшифрованию слов я заметил, что одинаковые буквы могут преобразовываться в разные и наоборот, что делает частотный анализ неуместным для взлома данного шифра, однако ключ для шифрования может повторяться, потому можно встретить повторы связок букв, по которым можно найти длину ключа, потом сам ключ и дешифровать сообщение (в виде шифртекста).

По самоключу можно сказать, что зная, что ключ это почти тот же шифртекст с добавлением буквы, можно перебрать все возможные буквы и найти ключевое слово. Аналогично, пробуя самоключ по открытому тексту, можно подбирать буквы и подставлять в алгоритм расшифровки.

6. Список использованных источников

1. https://yandex.ru/images/search?pos=0&img_url=https%3A%2F%2Fd3i71xaburhd42.cloudfront.net%2Fa290c7262acbec6969bb7408406fda47c9ab9036%2F5-Table1-1.png&text=англ%20алфавит%20с%20нумерацией%20с%200&lr=10743&rpt=simage&source=serp
2. https://ozlib.com/864764/informatika/shifr_vizhenera
3. https://www.familytree.ru/es/cipbooks/book021/history.htm
4. https://translated.turbopages.org/proxy_u/en-ru.ru.123aca93-62614cb4-00544a65-74722d776562/https/cryptography.fandom.com/wiki/Vigenère_cipher
5. https://ru.wikipedia.org/wiki/Индекс_совпадений#Индекс_совпадений