POLITECHNIKA ŚWIĘTOKRZYSKA	
Cyberbezpieczeństwo	Laboratorium nr 5
Numer ćwiczenia: 5	**Wykonał:** Karol Błędziński, Kamil Gorzała, Jakub Kołomański, Damian Zdyb

Cel:

1) Złamanie szyfru drugiej części zespołu używając programu Attack.py.
2) Wysłanie szyfru pierwszej części zespołu oraz opisanie działania szyfru Cipher.py.
3) Opisanie informacji na temat Auguste Kerckhoffs oraz konfuzje i dyfuzje.

Opis:

Auguste Kerckhoffs - był holenderskim językoznawcą i kryptografem, profesorem języków w paryskiej École des Hautes Études Commerciales w drugiej połowie XIX wieku.

1. System powinien być, jeśli nie teoretycznie, to w praktyce nie do złamania.
2. Projekt systemu nie powinien wymagać jego tajności, a ewentualne jego ujawnienie nie powinno przysparzać kłopotów korespondentom (zasada Kerckhoffsa).
3. Klucz powinien być: możliwy do zapamiętania bez notowania i dodatkowo – łatwy do zmienienia.
4. Kryptogramy powinny być możliwe do przesłania drogą telegraficzną.
5. Aparatura i dokumenty powinny być możliwe do przeniesienia i obsłużenia przez jedną osobę.
6. System powinien być prosty – nie wymagający znajomości wielu reguł ani nie obciążający zbytnio umysłu.

Zasada Kerckhoffsa – jedna z podstawowych zasad współczesnej kryptografii, sformułowana w XIX wieku przez holenderskiego kryptologa Augusta Kerckhoffsa. Zasada ta mówi, że system

kryptograficzny powinien być bezpieczny nawet wtedy, gdy wszystkie szczegóły jego działania – oprócz klucza – są znane.

Konfuzja - to ogół metod służących do ukrycia powiązań pomiędzy wiadomością jawną, szyfrogramem i kluczem.

Dyfuzja - umożliwia rozsianie bitów wiadomości jawnej i klucza w całej zawartości szyfrogramu. Większość standardowych szyfrów blokowych jest odporna na wszytkie znane formy kryptoanalizy, a wielkość klucza jest zwykle na tyle duża, że nie opłaca się mozolnie sprawdzać wszystkich możliwych kombinacji szyfru. Techniki obliczen kwantowych zapewniają interesujące możliwości, o których zapomina wielu użytkowników.

Analizowanie częstotliwości występowania znaków w szyfrogramie - to jeden z typów ataku ze znanym szyfrogramem. Polega na wyszukiwaniu często powtarzających się liter i popularnych sekwencji znaków.

We wszystkich językach różne litery używane są z różną częstotliwością. Dla każdego języka proporcje występowania poszczególnych znaków są nieco inne, więc teksty napisane w danym języku mają pewne wspólne właściwości, które pozwalają je odróżnić od tekstów napisanych w innych językach.

Przykładowo, w języku polskim często występują samogłoski takie jak a, e lub i. Z drugiej strony niezwykle rzadko zdarzają się niektóre spółgłoski, na przykład f lub ć. Istnieją zestawienia częstotliwości występowania liter w różnych językach. Dokładne rozkłady częstotliwości mogą się różnić w zależności od konkretnych rodzajów analizowanych tekstów (naukowych, prasowych, powieści i innych).

Każdy język posiada ponadto pewne typowe dla niego częste połączenia liter. Dla języka polskiego charakterystyczne są dwuznaki ch, cz, dz, dź, dż, rz i sz. Jest to cecha, która wyróżnia tekst w języku polskim od tekstów napisanych w innych językach. Dodatkowo, można dzięki temu lepiej przewidywać oryginalną kolejność liter z pomieszanych wyrazów.

Szyfr Cezara (zwany też szyfrem przesuwającym, kodem Cezara lub przesunięciem Cezariańskim) – jedna z najprostszych technik szyfrowania. Jest to rodzaj szyfru podstawieniowego, w którym każda litera tekstu jawnego

(niezaszyfrowanego) zastępowana jest inną, oddaloną od niej o stałą liczbę pozycji w alfabecie, literą (szyfr monoalfabetyczny), przy czym kierunek zamiany musi być zachowany. Nie rozróżnia się przy tym liter dużych i małych. Nazwa szyfru pochodzi od Juliusza Cezara, który prawdopodobnie używał tej techniki do komunikacji ze swymi przyjaciółmi.

Algorytm Vigenère'a – jeden z klasycznych algorytmów szyfrujących. Należy on do grupy tzw. polialfabetycznych szyfrów podstawieniowych. Szyfr ten błędnie został przypisany twórcy bardziej skomplikowanego szyfru Blaise'owi de Vigenère.

Szyfr, który obecnie nazywamy szyfrem Vigenère'a, po raz pierwszy został opisany przez Giovana Batista Belaso w 1553, w broszurze zatytułowanej La cifra del. Sig. Giovan Batista Belaso.

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

Jak można zauważyć, każdy z wierszy tablicy odpowiada szyfrowi Cezara, przy czym w pierwszym wierszu przesunięcie wynosi 0, w drugim 1 itd. Aby zaszyfrować pewien tekst, potrzebne jest słowo kluczowe. Słowo kluczowe jest tajne i mówi, z którego wiersza (lub kolumny) należy w danym momencie skorzystać.

Kod programu Brute Force Attack.py:

```
message = ''
LETTERS = 'ABCDEFGHIJKLMNOPQRSTUVWXYZ'

for key in range(len(LETTERS)):
    translated = ''

    for symbol in message:
        if symbol in LETTERS:
            num = LETTERS.find(symbol)
            num = num - key

            if num < 0:
                num = num + len(LETTERS)

            translated = translated + LETTERS[num]
        else:
            translated = translated + symbol

    print('Key #%s: %s' % (key, translated))
```

Wynik programu po wpisaniu szyfru „**BFMWU XMFMVM WXGOLQY**”:

```
Key #0: BFMWU XMFMVM WXGOLQY
Key #1: AELVT WLELUL VWFNKPX
Key #2: ZDKUS VKDKTK UVEMJOW
Key #3: YCJTR UJCJSJ TUDLINV
Key #4: XBISQ TIBIRI STCKHMU
Key #5: WAHRP SHAHQH RSBJGLT
Key #6: VZGQO RGZGPG QRAIFKS
Key #7: UYFPN QFYFOF PQZHEJR
Key #8: TXEOM PEXENE OPYGDIQ
Key #9: SWDNL ODWDMD NOXFCHP
Key #10: RVCMK NCVCLC MNWEBGO
Key #11: QUBLJ MBUBKB LMVDAFN
Key #12: PTAKI LATAJA KLUCZEM
Key #13: OSZJH KZSZIZ JKTBYDL
Key #14: NRYIG JYRYHY IJSAXCK
Key #15: MQXHF IXQXGX HIRZWBJ
Key #16: LPWGE HWPWFW GHQYVAI
Key #17: KOVFD GVOVEV FGPXUZH
Key #18: JNUEC FUNUDU EFOWTYG
Key #19: IMTDB ETMTCT DENVSXF
Key #20: HLSCA DSLSBS CDMURWE
Key #21: GKRBZ CRKRAR BCLTQVD
Key #22: FJQAY BQJQZQ ABKSPUC
```

Odpowiedź: PTAKI LATAJA KLUCZEM

Kod programu Cipher.py:

```python
# Your message
message = 'PTAKI LATAJA KLUCZEM'

key = 12 #Clue

mode = 'encrypt'

LETTERS = 'ABCDEFGHIJKLMNOPQRSTUVWXYZ' # Clue
# plaintext -> ciphertext or reversed
translated = ''
# capitalize the string in message
message = message.upper()

for symbol in message:
    if symbol in LETTERS:
        # get the encrypted (or decrypted) number for this symbol
        num = LETTERS.find(symbol) # get the number of the symbol
        if mode == 'encrypt':
            num = num + key
        elif mode == 'decrypt':
            num = num - key
        # wrap-around if num > length of LETTERS or less than 0
        if num >= len(LETTERS):
            num = num - len(LETTERS)
        elif num < 0:
            num = num + len(LETTERS)
        # add encrypted/decrypted number's symbol at the end of translated
        translated = translated + LETTERS[num]
    else:
        # just add the symbol without encrypting/decrypting
        translated = translated + symbol

print(translated)
```

Program działa na zasadzie szyfru przestawnego mający swoje początki w Erze Cezara. Polega on na przestawianiu liter o podany klucz który znała osoba rozszyfrowująca wiadomość.

Podsumowanie:

Laboratorium zakończyło się pomyślnie jedna grupa odczytała zaszyfrowaną wiadomość programem w którym użyto ataku typu brute force. Następnie zamieniliśmy się rolami i także sprawdziliśmy poprawność działania programów.