

**INSTYTUT MATEMATYKI I KRYPTOLOGII**

**Wydział Cybernetyki WAT**

**Przedmiot: ZAAWANSOWANA  
MATEMATYKA DYSKRETNA**

**SPRAWOZDANIE**

**Temat: PROTOKÓŁ WYMIANY GGH**

**Wykonał:**

**Paweł Witkowski,  
K5X4S1  
[nr. 64024]**

**Data wykonania ćwiczenia:**

**01.02.2017**

**Prowadzący ćwiczenie:**

**Kpt. Dr Mariusz Jurkiewicz**

## 1. Wstęp teoretyczny

Protokół opierający swoje bezpieczeństwo na problemie znalezienia najkrótszego wektora oraz najbliższego (SVP,CVP).

Generowana jest baza  $\mathbf{B}$ , na której będzie oparta struktura kraty  $\mathbf{L}$ , co dalej będzie oznaczane przez  $\mathbf{L}(\mathbf{B})$ . Współczynniki wektorów  $\mathbf{B}$  ( $v_1, v_2 \dots v_n \in \mathbb{Z}^n$ ), która będzie pełnić rolę klucza prywatnego, powinny być wyznaczone jednostajnie, dla zadanego  $\mathbf{d}$  ze zbioru  $\{-d, \dots, 0, 1, \dots, d\}$ .  $\mathbf{B}$  powinna być możliwie najbardziej ortogonalna, znaczy prostopadła, co można określić licząc tzw. "Współczynnik Hadamara" określony wzorem:

$$H(\mathbf{B}) = (\text{Det}(\mathbf{L}) / (\|v_1\| \|v_2\| \dots \|v_n\|))^{1/n}, \quad H(\mathbf{B}) = (0, 1]$$

Gdzie dla kraty pełnego rzędu:

$$\text{Det}(\mathbf{L}) = \text{Det}(\mathbf{B}).$$

Im bardziej ortogonalna jest baza  $\mathbf{B}$ , tym bardziej  $H(\mathbf{B})$  jest bliższe jedności.

Następnie należy skorzystać z twierdzenia,  $L(\mathbf{B}) = L(\mathbf{B}')$ , przy założeniu, że

$$(\mathbf{B}')^T = \mathbf{A} (\mathbf{B})^T, \text{ gdzie } \mathbf{A} \text{ to macierz unimodularna}$$

Jest to istotne ponieważ, możemy wyznaczyć klucz publiczny. By to zrobić, trzeba stworzyć odpowiednią zdeformowaną macierz unimodularną  $\mathbf{A}$ , która posłuży do zdeformowania klucza prywatnego.

$$\mathbf{A} = \mathbf{U}_1 * \mathbf{U}_2 * \mathbf{U}_3 \dots \text{do momentu gdy } H(\mathbf{A}) < 0,1$$

, gdzie  $\mathbf{U}_i$  to macierze trójkątne z zerami pod/nad przekątną. Lub macierze trójkątne na których wykonano działania elementarne, które nie zmieniają wyznacznika.

Właściwa część protokołu wymiany danych GGH, polega na stworzeniu klucza prywatnego oraz publicznego, przez potencjalnego odbiorcę, oraz nadanie klucza publicznego do sieci czyli również do nadawcy. Kluczem publicznym jest właśnie zdeformowana baza  $\mathbf{B}$ , czyli  $\mathbf{B}'$  o wektorach ( $v'_1, v'_2 \dots v'_n \in \mathbb{Z}^n$ ).

Nadawca wiadomości pobiera klucz publiczny, mnoży swój wektor wiadomości  $\mathbf{m}$ , którą chce wysłać przez  $\mathbf{B}'$ , prawostronnie. Następnie dodaje do tego wektor błędu  $\mathbf{r}$ , o wielkości  $\mathbf{n}$ . Współczynniki  $\mathbf{r}$  powinny być odpowiednio małe.

$$\mathbf{c} = \mathbf{mB}' + \mathbf{r}$$

Tak zadane  $c$ , jest wysyłane z powrotem do nadawcy klucza publicznego. Odbiorca wiadomości mnoży szyfrogram przez  $B^{-1}$  lewostronnie. Dzięki temu otrzymuje współczynniki kombinacji liniowej użyte do pomnożenia kraty. Następnie należy otrzymany wektor pomnożyć przez odwrotny klucz publiczny prawostronnie. Wynikiem jest odszyfrowana wiadomość. Protokół zapewnia bezpieczeństwo, ponieważ strona podsłuchująca, jeśli użyje algorytmu Babaja na podstawie klucza publicznego otrzyma kompletnie inny wektor, co obrazuje:

$$\|v - w\| \lll \|v' - w\|$$

## 2. Teoria w praktyce

- o Użytkownik, dalej nazywany Alicją, chcący odebrać wiadomość podaje rozmiar jego macierzy oraz wartość  $d$ .
- o Z takiego zakresu jest generowana dla niego  $B$  tj klucz prywatny, o współczynniku Hadamara większym niż 0,8.
- o Generowana jest macierz unimodularna  $U$  o współczynniku Hadamara mniejszym niż 0,1.
- o Generowany jest klucz publiczny  $P$ , na podstawie  $P^T = U B^T$ .
- o Alicja wysyła klucz publiczny do sieci, odbiera go nadawca wiadomości, dalej nazywany Bobem.
- o Bob wprowadza swoją wiadomość jako wektor
- o Następuje wymnożenie wiadomości przez klucz publiczny prawostronnie, oraz dodanie możliwie małego wektora błędu. Taka wiadomość jest wysłana do Alicji.
- o Alicja używa algorytmu Babaja na szyfrogramie używając swojego klucza prywatnego. Otrzymany wektor dalej będzie nazywany szyfrogramem.
- o Odwrócenie klucza publicznego.
- o Prawostronne wymnożenie szyfrogramu przez klucz publiczny odwrócony.
- o Otrzymanie odszyfrowanej wiadomości.

### 3. Wyniki działania programu

Program jest symulatorem protokołu wymiany danych. Z tego powodu jego forma przybrała taką postać:

Dla  $n=5$ , przedział  $\{-d \dots 0, 1, 2 \dots d\}$  jest większy. Poprawne odszyfrowanie wiadomości.

```
E:\studia\sem3\zmd\GGH\Debug\GGH.exe
Podaj ilosc wektorow oraz rozmiar przestrzeni
Dla prokolu GGH musi to byc krata pelnego rzędu.
5

-----
Alicja
Podaj 'd' z zakresu ktorego bedzie tworzony klucz prywatny
200
Klucz prywatny wygenerowany dla Alicji, o wspolczynniku Hadamara:0.817618
-164 -54 -34 126 157
8 -29 63 -149 -129
-171 122 -151 -3 20
149 28 43 -191 143
-80 -19 157 76 69
Wygenerowany klucz publiczny o wspolczynniku Hadamara:0.0453673
-164 -54 -34 126 157
172 25 97 -275 -286
6829 1984 2159 -7381 -8380
-12622 -3461 -3763 11915 14288
18283 6130 6959 -26704 -28337

-----
Bob
Podaj liczby do zaszyfrowania
87
65
22
168
-5

Wiadomosc do zaszyfrowania:
87 65 22 168 -5

-----
Alicja
Otrzymałem szyfrogram:
-2064761 -571522 -616133 1965944 2352779
Wspolczynniki kombinacji liniowej, wyznaczone algorytmem Babaja:
12476 -2858 120 163 -5
Po zdeszyfrowaniu szyfrogramu, otrzymane 'm':
87 65 22 168 -5
```

n=7, d=150, nadal poprawne wyniki.

```
E:\studia\sem3\zmd\GGH\Debug\GGH.exe
Podaj ilosc wektorow oraz rozmiar przestrzeni
Dla prokolu GGH musi to byc krata pelnego rzędu.
7
-----
Alicja
Podaj 'd' z zakresu ktorego bedzie tworzony klucz prywatny
150
Klucz prywatny wygenerowany dla Alicji
71 -115 64 -116 -70 27 -99
45 65 -90 -105 150 -70 87
-11 63 132 -148 83 -129 -133
-38 57 44 -150 -125 -57 51
66 28 54 150 -144 -112 109
-67 9 -89 52 -32 -93 52
102 118 44 53 -121 -77 -90
Wygenerowany klucz publiczny
71 -115 64 -116 -70 27 -99
45 65 -90 -105 150 -70 87
1364 -2302 1502 -2363 -1467 481 -2200
-2616 4739 -2944 4102 3122 -1261 4393
-2685 7415 -1876 -335 5258 -4812 4394
7430 -21161 7719 -2127 -16638 12464 -14305
-6548 23416 -5441 -4334 18002 -17286 13256
-----
Bob
Podaj liczby do zaszyfrowania
150
65
121
0
-50
68
516
Wiadomosc do zaszyfrowania:
150 65 121 0 -50 68 516
-----
Alicja
Otrzymałem szyfrogram:
-2560658 9981392 -2003371 -2674379 7716491 -7773723 5372262
Wspolczynniki kombinacji liniowej, wyznaczone algorytmem Babaja:
-48098 31094 23629 13688 3358 584 516
Po zdeszyfrowaniu szyfrogramu, otrzymane 'm':
150 65 121 0 -50 68 516
```

n=8, d=100, nadal poprawne wyniki.

```
E:\studia\sem3\zmd\GGH\Debug\GGH.exe
Podaj ilosc wektorow oraz rozmiar przestrzeni
Dla prokolu GGH musi to byc krata pelnego rzędu.
8
-----
Alicja
Podaj 'd' z zakresu ktorego bedzie tworzony klucz prywatny
100
Klucz prywatny wygenerowany dla Alicji
99 -8 85 -31 83 -47 45 -91
57 8 6 75 37 -74 -87 -90
40 32 -78 56 -52 -13 96 -46
-87 22 56 -29 -47 -29 34 38
-41 -84 24 87 -94 68 63 96
-7 93 39 96 51 35 -75 70
92 -37 -66 89 13 -35 81 -62
-51 -97 -11 20 -43 -43 85 63
Wygenerowany klucz publiczny
99 -8 85 -31 83 -47 45 -91
-339 40 -334 199 -295 114 -267 274
577 -24 426 -205 409 -221 453 -502
105 150 -245 395 -175 -243 16 -279
-2578 526 -2935 2028 -2839 681 -1341 1899
1481 -239 1511 -1026 1736 -234 401 -954
1071 218 -124 295 202 -256 1229 -855
4270 -1772 5984 -5302 4842 -502 4681 -2549
-----
Bob
Podaj liczby do zaszyfrowania
123
564
10
-50
65
11
0
-8
Wiadomosc do zaszyfrowania:
123 564 10 -50 65 11 0 -8
-----
Alicja
Otrzymalem szyfrogram:
-363938 59573 -383436 249573 -347506 114161 -261525 285606
Wspolczynniki kombinacji liniowej, wyznaczone algorytmem Babaja:
-4374 1198 380 125 54 35 -16 -8
Po zdeszyfrowaniu szyfrogramu, otrzymane 'm':
123 564 10 -50 65 11 0 -8
```

```
E:\studia\sem3\zmd\GGH\Debug\GGH.exe
Podaj 'd' z zakresu ktorego bedzie tworzony klucz prywatny
50
Klucz prywatny wygenerowany dla Alicji
-16 -14 -40 13 -29 31 21 26 -2
-11 -32 -36 -23 48 -1 -28 30 1
-15 50 -49 -36 0 33 -26 2 9
16 49 19 -20 -15 22 -9 32 -27
-37 31 -13 -30 38 -12 -5 46 24
-47 -11 23 -27 12 0 -46 22 35
11 35 23 -5 44 4 8 -2 3
4 10 -25 5 48 -37 35 -13 1
-25 7 37 -45 -24 -37 -8 -8 23
Wygenerowany klucz publiczny
-16 -14 -40 13 -29 31 21 26 -2
-11 -32 -36 -23 48 -1 -28 30 1
27 42 39 -134 212 -93 -166 -42 19
167 281 328 -415 612 -348 -514 -232 34
131 341 327 -487 723 -411 -578 -268 98
-2 107 163 -114 168 -163 -178 -224 118
-52 212 227 -453 745 -399 -582 -330 283
82 219 263 -647 1323 -715 -819 -573 373
-380 -309 -337 -100 -67 230 -83 469 128

-----
Bob
Podaj liczby do zaszyfrowania
123
99
21
912
-87
-666
0
517
56

Wiadomosc do zaszyfrowania:
123 99 21 912 -87 -666 0 517 56

-----
Alicja
Otrzymałem szyfrogram:
135013 247253 271562 -603779 1069232 -528076 -731681 -303776 144155
Wspolczynniki kombinacji liniowej, wyznaczone algorytmem Babaja:
0 0 0 0 0 0 0 0 0
Po zdeszyfrowaniu szyfrogramu, otrzymane 'm':
0 0 0 0 0 0 0 0 0
```

n=9, d=50 brak poprawności.

```
E:\studia\sem3\zmd\GGH\Debug\GGH.exe
Dla prokolu GGH musi to byc krata pelnego rzędu.
9
-----
Alicja
Podaj 'd' z zakresu ktorego bedzie tworzony klucz prywatny
15
Klucz prywatny wygenerowany dla Alicji
-10 -3 13 -4 12 -15 5 -15 1
9 -5 -2 13 5 2 -3 -12 7
10 -8 -3 -12 5 -14 -2 14 14
15 -1 -9 -3 13 -10 -12 8 -10
-5 -5 -7 -14 -15 -2 -7 0 5
10 -5 7 4 -2 7 -3 -6 7
2 5 -10 -8 13 13 2 -14 5
-14 -7 -7 -8 -14 9 -5 -15 1
-9 -9 6 -15 -10 5 15 14 3
Wygenerowany klucz publiczny
-10 -3 13 -4 12 -15 5 -15 1
39 4 -41 25 -31 47 -18 33 4
31 6 -40 -13 -36 29 -14 71 4
-7 -1 19 39 37 -8 -4 -74 -22
-69 -22 66 -56 32 -83 31 -57 27
7 13 16 -6 36 -16 -7 5 -45
-81 7 80 56 89 -24 51 -184 -12
110 -2 -118 26 -89 114 -65 109 11
-62 28 32 74 93 47 30 -187 -82
-----
Bob
Podaj liczby do zaszyfrowania
5
12
32
96
51
0
97
-8
-99
Wiadomosc do zaszyfrowania:
5 12 32 96 51 0 97 -8 -99
-----
Alicja
Otrzymałem szyfrogram:
-5379 -3069 9019 -1351 3857 -11478 3055 -7624 6311
Wspolczynniki kombinacji liniowej, wyznaczone algorytmem Babaja:
570 107 186 -258 152 -6 -101 -107 -99
Po zdeszyfrowaniu szyfrogramu, otrzymane 'm':
5 12 32 96 51 0 97 -8 -99
```

n=9,d=15 poprawne wyniki.



#### 4. Wnioski

Protokół wymiany działa poprawnie dla odpowiednich liczb. Odpowiednie liczby oznaczają, nie za duże. Im większy jest wymiar macierzy tym mniejsze muszą być wartości podawane do klucza prywatnego. Wszelkie błędy wynikające z odczytu są spowodowane małym zakresem typu Long Double. Pojawiające się zera oznaczają, że liczba się “przekreśliła”. Możliwe jest zwiększenie macierzy do  $n=10$ , aczkolwiek wartości musiałyby być zmniejszone do  $d=(3,5)$ , co znacząco zmniejszyłoby skuteczność rozwiązania. Jest to spowodowane stosunkowo dużym, dla tak małego  $d$ , wektora błędu. Dla niektórych wartości poda poprawny wynik, dla niektórych nie. Ryzyko błędu zdecydowanie przekreśla sens użycia większych macierzy.

Bezpieczeństwo algorytmu opiera się na SVP oraz CVP. Zaatakowanie go przy pomocy klucza publicznego, powoduje odnalezienie kompletnie innego wektora. Również należącego do  $L(\mathbf{B})$ , lecz w dużo większej odległości niż jest to możliwe do odczytania wiadomości. Problemem algorytmu jest przekazywanie informacji o kracie.

Przejsie z typu Long Double na `__int64` nie przyniosło poprawy, a wręcz przeciwnie. Protokół był dużo mniej efektywny, ponieważ wyznacznik macierzy szybciej się zerował. Jest to prawdopodobnie spowodowane sposobem przechowywania liczby w tych dwóch typach. Na potrzeby przykładu został użyty typ Long Double, co jest bardzo złym wyborem, ponieważ działania na liczbach zmiennoprzecinkowych są pozbawione dokładności. Jest to nadrobione rzutowaniem na tym całkowity, co redukuje znacząco szansę pomyłki, ale nie można jej wykluczyć. Najlepszym rozwiązaniem byłoby wykorzystanie bibliotek typu BigInteger, które zachowałyby dokładność i byłyby “prawie” odporne na przekreślenia się liczby.