Bezpieczeństwo komputerowe Lista 3

Radosław Wojtczak

Numer indeksu: 254607

0.1 Wprowadzenie

Celem tej listy było przetestowanie w praktyce jakie niebezpieczeństwo niesie ze sobą wykorzystywanie tego samego klucza do kodowania wielu wiadomości następującym sposobem:
$c_i = m_i \oplus G(k)$.

0.2 Omówienie rozwiązania

Na podstawie 20 otrzymanych kryptogramów należało zdeszyfrować kryptogram nr 21. Aby tego dokonać wykorzystana została wskazówka przedstawiona przez Pana Profesora w trakcie wykładu:
$c_i \oplus c_j = m_i \oplus G(k) \oplus m_j \oplus G(k) = m_i \oplus m_j$. Od tego momentu dokonujemy analizy otrzymanego XOR'a dwóch tekstów jawnych w oparciu o znajomość kodowania ASCII. Wiemy, iż kod spacji zapisany na 8 bitach wynosi *00100000*, natomiast każda z liter ma "zapalony" bit na pozycji 64 (*01xxxxxx*). Automatycznie zauważamy, iż XOR spacji oraz litery nie powoduje zmiany drugiego najbardziej znaczącego bitu, więc możemy się spodziewać, że jeden z tekstów jawnych w danej pozycji ma spację (oczywiście, jest to lekkie uproszczenie ze względu na to, iż pozostałe znaki przestankowe takie jak {.,!?"} również mają kody poniżej 64, aczkolwiek są znacznie rzadziej używane niż spacje), a drugi literę. Dodatkowo zauważamy, że XOR litery ze spacją jedynie zmienia jej wielkość (różnica między małą literą a wielką to 32, przykład: $a = 01\mathbf{1}00001, A = 01\mathbf{0}00001$), więc jeśli jeden z tekstów w danym miejscu posiada spację to drugi automatycznie musi posiadać literę o "przeciwnej wielkości" względem litery reprezentowanej przez XOR ów tekstów jawnych.
Szybko zauważamy, że posiadając jedynie dwa kryptogramy taki sposób rozwiązywania zadanie mógłby być dość nieefektywny, gdyż na każdej wyżej wspomnianej pozycji musielibyśmy rozpatrzeć dwa przypadki: m_i jest tekstem jawnym ze spacją, a m_j z literą oraz przypadek odwrotny. Zauważamy, że wtedy otrzymujemy 2^k możliwych kluczy, gdzie **k** reprezentuję liczbę takich miejsc. Tutaj do gry wkraczają dodatkowe kryptogramy, które zdecydowanie usprawniają pozwyższe rozumowanie.
Ustalmy dany kryptogram i oznaczmy go przy pomocy l_1. Wtedy zauważamy, że dokonując XORa l_1 z dowolnym l_x otrzymujemy xor tekstów jawnych m_1 i m_x, $x \in 1, ..., n$, gdzie **n**=liczba otrzymanych kryptogramów. Od tego momentu dla każdego ustalonego kryptogramu generujemy specjalny licznik, który zlicza ile razy dla danego indeksu wykryliśmy, że jeden z kryptogramów zawiera tam spację, przy dokonywaniu XOR'ów powyższych par. Biorąc pod uwagę, że jeden z nich z góry został przez nas ustalony zakładamy, że nie może być przypadkiem, gdy przykładowo na 19 xorów na pozycji o indeksie 10 spacja wystąpiła 19 razy- dochodzimy do wniosku, że kryptogram l_1 w tym miejscu posiada spację.
Następnie ustalamy kryptogram l_2 i dokonujemy analogicznego rozumowania, aż nie rozpatrzymy wszystkich kryptogramów.

0.3 Prawdopodobieństwo spacji

Załóżmy, że indeks **x** jest indeksem tekstu jawnego **m**. Podejrzewamy, że w tym miejscu znajduje się spacja. Szybko zauważamy, iż porównanie tego indeksu

z tym samym indeksem dla następnych **n-1** kryptogramów i sprawdzenie, czy test spacji przeszedł **n-1** razy może być zbyt surowym kryterium. Biorąc pod uwagę liczbę rozpatrywanych kryptogramów może dojść do sytuacji, w której dla tego samego indeksu w dwóch tekstach jawnych znajdują się dwa znaki przestankowe, lub inne symbole, których kod ASCII jest mniejszy niż 64. Wtedy przestajemy rozpatrywać ów indeks, jako potencjalny indeks spacji z fałszywej pobudki. Oczywiście taka stronniczość może zajść w drugą stronę- jako spację ustalimy indeks, gdzie tak naprawdę w tekście jawnym znajdowała się na przykład kropka. Dochodzimy do wniosku, że poszukiwanie spacji jest bardzo podatne na wyniki fałszywo-dodatnie jak i fałszywo-ujemne, dlatego dodatkowo wprowadzamy **Prawdopodobieństwo spacji-** jest to liczba z przedziału **1...n-1** od której uznajemy, że dany indeks zawiera spację.

0.4 Wyniki i interpetacja

Po wykonaniu wielu prób udało się ostatecznie zdekodować otrzymany kryptogram. Jego treść: **Duma Kaczyńskiego. Namówił bojówki do bicia kobiet, a ksiądz "chce je powystrzelać".** Poniższa tabela przedstawia jak liczba kryptogramów(**l**) i prawdopodobieństwo spacji(**s**) wpływa na otrzymany tekst.

l	s	Tekst
20	14	Fum* Kaczynskiego. N*mowil baj*w*i do bicia kobiet, a ksiadz "chce je powystrzela*"
20	16	Fum* Kaczy*skiego.***mowil*baj***i*d* bici* k*biet, a *sia*z *chc* je po*yst*z**a*"
20	19	**** K*******i*********************************i*** * ************ ***************"
20	12	[uma Kaczynskiego;-Namowil bajoz*i do bicio kobiet, a ksiadz "|hce ji powystrzela*"
19	14	Fum* Kaczynskiego. N*mowil baj*w*i*do bici* kobiet, a ksia*z "chc* je powystrz**a*"
19	16	Fum* Kaczy*skiego.***mow*l*boj******* bici**k*bi*t* a *sia*z *chc* je p**yst*z**a*"
19	18	**** K*******i*********************************i*** * ************ ***************"
19	12	[uma Kaczynskiego;-Namowil bajow*i do.bicio kobiet, a ksiadz "chce ji powystrzela*"
19	10	[uma Kaczynskiego;-Namowil bajow*i do.bicio kobiet, a ksiadz "ch|e ji powystrzela*"
18	15	Fum* Kaczy*skiego.***mow*l*boj******* bici**k*bi*t, a *sia*z *chc* je p**yst*z**a*"
18	13	Fum* Kaczynskiego. Namowil baj*w*i*do bici* kobiet, a ksia*z "chce je powystrz**a*"
17	15	**m* **cz****i***.***mow*l**o******** bici****bi****a *s*a****chc* *e*p***st****a*"
17	13	Fum* *aczy*skie*o.*N*mowil baj*w*i*d* bici* k*biet,*a *sia*z "chce je po*yst*z**a*"
17	11	[uma *aczynskie*o; Namowil bajpw*i do.bicio kobiet,*a ksiadz "chce ji |owystrzela*"

Table 1: Otrzymane wyniki

Interpretacja: Zauważamy, iż nie udało nam się odzyskać całkowicie klucza, jednakże dla danych $l = 20$ oraz $s = 14$ otrzymujemy tekst, który bez problemu potrafimy odpowiednio rozszyfrować ze względu na nieliczne ubytki klucza lub złe dopasopwanie liter. Ów złe dopasowanie liter (takie jak wystąpienie F zamiast D w wyrazie Duma) wynika z faktu, iż fałszywie założyliśmy wystąpienie spacji w tym miejscu (prawopodobnie znajdował się inny znak interpunkcyjny taki jak ").
Ponadto widzimy, że ze zmniejszeniem się licbzy kryptogramów spada nasza możliwość rozkodowania otrzymanej wiadomości. Dodatkowo prawdopodobieństwo spacji ustalone na liczbę $n-1$ daje bardzo złe wyniki (z wyżej opisanego powodu)

i zauważamy, że ustalenie tej liczby na mniej więcej 70% liczby wprowadzonych kryptogramów skutkuje najlepszymi rezultatami.
Kolejne czynniki, które wpływają na dokładność otrzymanego klucza to długość kryptogramów (która wpływa dość znacznie na naszą umiejętność rozkodowania wiadomości) oraz kolejność czytania kryptogramów (której wpływ zależy od progu prawdopodobieństwa spacji).

0.5 Wnioski

Wykonane doświadczenia pokazują, że niewielkim nakładem sił przy dzisiejszej mocy obliczeniowej komputerów jesteśmy w stanie z łatwością wykorzystać niepoprawne stosowanie kluczów jednorazowego użytku (OTP, one-time pad). Przedstawione wyżej rozwiązanie nie gwarantuje otrzymania dokładnie odkodowanego kryptogramu, jednak dokładność jest na tyle duża aby zrozumieć przesłaną informację. Użycie dodatkowych metod bazujących na znajomości struktury przesłanego kryptogramu (np. *frequency attack*) może zwiększyć nasze szanse na automatyczne odkowodowanie wiadomości.