Bezpieczeństwo komputerowe

Laboratorium - lista nr 2, 22 X

Zrób jedno z poniższych zadań

Zadanie 1 (10 pkt) Przechwyciłaś/eś kilkanaście kryptogramów. Wiesz, że każdy z nich powstał jako rezultat szyfrowania wiadomości za pomocą szyfru strumieniowego. Co więcej, do szyfrowania każdej wiadomości wykorzystano ten sam klucz, czyli: $c_i = \mathsf{Enc}(\mathsf{k}, \mathsf{m_i}) = m_i \oplus G(k)$ dla $i = 1 \dots l$, gdzie G jest generatorem bitów pseudolosowych, a k jest kluczem tajnym.

Napisz program (i umieść go na swoim koncie na github), który przyjmuje na wejściu l kryptogramów zaszyfrowanych za pomocą szyfru strumieniowego z tym samym kluczem. Na wyjściu program ma zwrócić teksty jawne.

Przeprowadź eksperymenty, aby określić skuteczność programu w zależności od:

- długości kryptogramów,
- liczby kryptogramów dla $l = 1, 2, 3, 4, \dots$ (od jakiej wartości l, program zaczyna działać?)
- typu szyfru strumieniowego (Salsa20, Sosemanuk, ...)
- wykorzystanego kodowania znaków ASCII/UTF-8/ISO-8859-2?

Aby uzyskać przykładowe dane, wprowadź numer indeksu do formatki na stronie.

Zadanie 2 (10 pkt) Podany kryptogram został wygenerowany za pomocą szyfru AES z kluczem o długości $n = 256$-bitów, w trybie CBC. Napisz program, który przeszukuje całą przestrzeń kluczy i deszyfruje podany kryptogram.

Klucz został wygenerowany w następujący sposób (NIGDY tak nie generuj kluczy!):

jest wynikiem obcięcia (do pierwszych 16 znaków) ciągu będącego hashem pewnej wartości $key \leftarrow SHA256(sekret).substr(0, 16)$.

Jako dane dla Twojego programu otrzymujesz:

- kryptogram,
- użyte IV,
- k_2 będące sufiksem key (tj. $key = k_1 k_2$ – key jest konkatenacją k_1 i k_2),
- określenie podprzestrzeni kluczy k_1, które mają zostać sprawdzone.

Uruchom program na tylu rdzeniach ile wspiera Twój komputer.

Ile kluczy sprawdza Twój program (porównaj worst-case, average case)?

Jaka jest oczekiwana liczba "sekretów", którą należałoby wygenerować, aby uzyskać dla dwóch różnych wartości ten sam klucz?

Ile kluczy musiałby sprawdzić program gdyby klucz powstawał jako wartość prawdziwie losowa?

Jaki byłby koszt złamania w ciągu 24-godzin $n - k = 48/56/64/80/100$-bitów (spróbuj oszacować koszty bazując na wynikach osiąganych na Twoim komputerze i cenach rozwiązań chmurowych)?