Andrzej M. Borzyszkowski

Tryby szyfrów blokowych

Na wykładzie, slajdy 6-12, przedstawiono kilka trybów szyfrów blokowych, z czego nas będzie interesować w tej chwili tryb EBC, książki kodowej, oraz tryb CBC, dodawania zaszyfrowanego bloku do szyfrowania kolejnego bloku. Pierwszy tryb jest zbyt prosty, by być odporny na wielokrotne szyfrowanie, czy na atak z tekstem jawnym. Celem tego ćwiczenia będzie unaocznienie tej prostoty w sposób wizualny.

W zadaniu, należy zaprojektować "szyfrowanie" obrazu graficznego. Obraz powinien być czarno-biały i mieć rozmiar rzędu kilkuset pikseli w pionie i w poziomie. Obraz taki należy podzielić na małe bloki, np. 8x8 pikseli, w ten sposób każdy blok grafiki zostaje potraktowany jako blok szyfru blokowego. Cały obraz należy potraktować jako ciąg małych bloków, np. przeglądanych kolejnymi wierszami. W naszym przypadku nie dysponujemy własną implementacją szyfru blokowego, w celu wykonania zadania można przyjąć dowolne przekształcenie, bez koniecznosci "odszyfrowywania" kryptogramu, jest istotne, by takie same bloki były identycznie szyfrowane. Np. można zastosować jakąkolwiek funkcję skrótu, md5sum czy sha1sum.

Uwaga: celem zadania jest zrozumienia działania trybów blokowych, a nie tylko ich nazw. W związku z tym w rozwiązaniu **nie można** stosować gotowych bibliotek wywołujących szyfrowanie w jakimkolwioek trybie szyfru blokowego.

Program powinien wczytać plik graficzny i wyprodukować dwa pliki graficzne: kryptogram zaszyfrowany w trybie ECB oraz kryptogram zaszyfrowany w trybie CBC. Należy pamiętać, że obrazek powinien być maksymalnie nieskomplikowany, np. jakiś znak firmowy albo powiększona do dużych rozmiarów czcionka. Przykłady: 1.1_o jest przekształcany w trybie 1.2_ecb oraz 1.3cbc i drugi 2.1_o, 2.2_ecb i 2.3_cbc.

Zadanie:

Program block powinien czytać pliki: graficzny plain.bmp i opcjonalnie tekstowy key.txt z kluczem i powinien zapisywać dwa pliki graficzne "zaszyfrowanego" obrazu ecb_crypto.bmp oraz cbc_crypto.bmp. W rozwiązaniu należy przesłać program w wersji źródłowej i skompilowanej jak również testowy plik graficzny i ew. plik z kluczem.
Uwaga: w programie **nie wolno** stosować wbudowanych bibliotek kryptograficznych z wywołaniem funkcji szyfrowania w wybranym trybie blokowym, należy te tryby zaimplementować własnoręcznie.