Podstawy kryptografii

Andrzej M. Borzyszkowski

Instytut Informatyki
Uniwersytet Gdański

sem. letni 2023/2024

inf.ug.edu.pl/~amb/

Szyfry blokowe

Funkcje pseudolosowe

- Ozn.: $[n]$ – zbiór ciągów bitów długości n, $\oplus$ dodawanie bitów mod 2
- Dana funkcja $F : [n] \times [n] \to [n]$, $F(k, m)$ – klucz i wiadomość
  – obliczalna PPT
  – problem, czy funkcję jednej zmiennej $F(k, \_)$ da się odróżnić od losowej funkcji $f : [n] \to [n]$?
  – funkcji pierwszego typu jest tylko 2^n, drugiego typu $(2^n)^{2^n}$
  – sam zapis funkcji $f : [n] \to [n]$ wymaga wykładniczych zasobów
  – jeśli f jest losowa, to wartości $f(x)$ oraz $f(y)$, $x \neq y$, są niezależne
- Funkcja F jest pseudolosowa, jeśli przeciwnik o zasobach PPT nie odróżni $F(k, \_)$, k losowe, od losowej funkcji f
  – nawet nie zna całej funkcji
  – korzysta jedynie z wyroczni obliczającej wartości funkcji
  – tzn.: podobnie jak w ataku z wybranym tekstem jawnym

Wykorzystanie funkcji pseudolosowych

- Metoda naiwna
  – $Enc(k, m) = F(k, m)$
  – nie ujawnia niczego o tekście
  – ale jest deterministyczna, nie może być bezpieczna przeciwko atakowi z wybranym tekstem jawnym czy z zestawem tekstów jawnych
- Metoda słuszna
  – generowanie klucza: losowy wybór
  – funkcja szyfrowania: $Enc(k, m) = \langle r, F(k, r) \oplus m \rangle$, r jest losowo wybranym ciągiem $r \in [n]$
  – funkcja odszyfrowania: $Dec(k, \langle r, c \rangle) = F(k, r) \oplus c$
  – r nie może pojawić się powtórnie, prawdopod. tego błędu jest zaniedbywalnie małe
- Tw.: jeśli funkcja jest pseudolosowa, to powyższy szyfr jest bezpieczny przeciwko atakowi z wybranym tekstem jawnym

Permutacje pseudolosowe

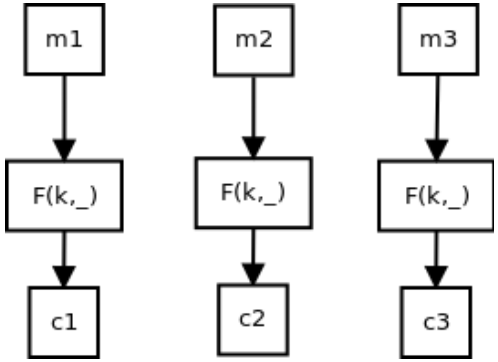
- Funkcja $F : [n] \times [n] \to [n]$, jest permutacją z kluczem, jeśli każda $F(k, \_)$ jest bijekcją
  – jest efektywna, jeśli istnieją algorytmy PT obliczające $F(k, \_)$ oraz funkcję odwrotną
- F jest pseudolosową permutacją, jeśli jest nieodróżnialna od losowych permutacji
  – równie dobrze jest nieodróżnialna od losowych funkcji
- Można żądać więcej, by $F(k, \_)$ oraz funkcja odwrotna były nieodróżnialne od losowych permutacji

Tryby szyfrów blokowych

- Konieczność szyfrowania tekstów jawnych dłuższych niż blok
  – tekst jawny jest ciągiem bloków P_j, szyfrogram bloków C_j, $j = 1, 2, \ldots$
  – być może trzeba wypełniać ostatni blok
  - standardowa metoda:
  jeśli zabraknie jednego bajta, to wpisujemy ascii 1
  jeśli dwóch, to dwa bajty ascii 2, itd.
  jeśli nie ma potrzeby uzupełniania, to dodajemy cały blok z ascii 0
  gdyby nie dodawać pustego bloku, to nie odróżnilibyśmy np. tekstu jawnego z ostatnim bajtem równym 1 od tekstu o jeden bajt krótszego

Tryby: ECB

- ECB (*electronic code book*)
  – każdy blok szyfrowany jest niezależnie:
  $c_j = Enc(k, m_j)$, $m_j = Dec(k, c_j)$
  – daje to szyfr deterministyczny $Enc(k, m) = F(k, m)$

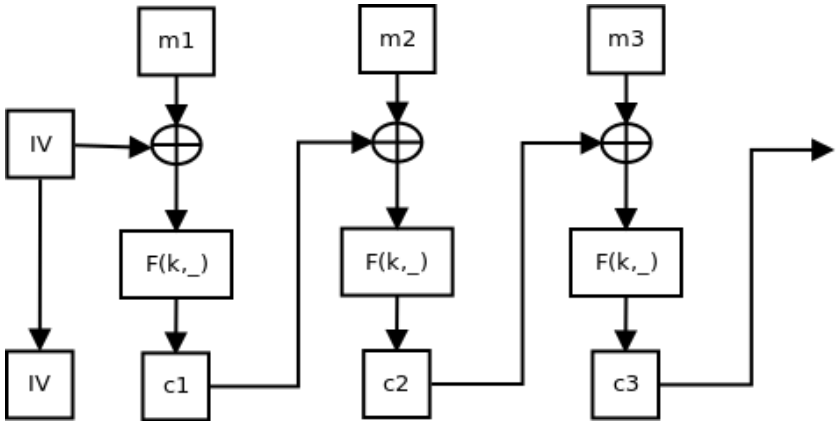


Własności ECB

- Jest to szyfr deterministyczny $Enc(k, m) = F(k, m)$
  – gdyby $Enc(k, m) = \langle r, F(k, r) \oplus m \rangle$
    - trzeba by dużej liczby losowych wektorów,
    - szyfrogram byłby dwa razy dłuższy niż tekst jawny
- Zalety:
  – niezawodność,
  – łatwość szyfrowania fragmentów (baza danych, zawartość dysku)
- Wady:
  – brak bezpieczeństwa każdego rodzaju, Ewa może rozpoznać, która z wiadomości została zaszyfrowana
  – niebezpieczeństwo rozpoznania stałych fragmentów, niebezpieczeństwo manipulacji fragmentami szyfrogramu
  – nie powinien być stosowany w ogóle

Tryby: CBC

- CBC (*cipher block chaining*)
  – $c_j = Enc(k, m_j \oplus c_{j-1})$, $m_j = Dec(k, c_j) \oplus c_{j-1}$, $c_0 = IV$
  – takie same fragmenty tekstu są szyfrowane odmiennie

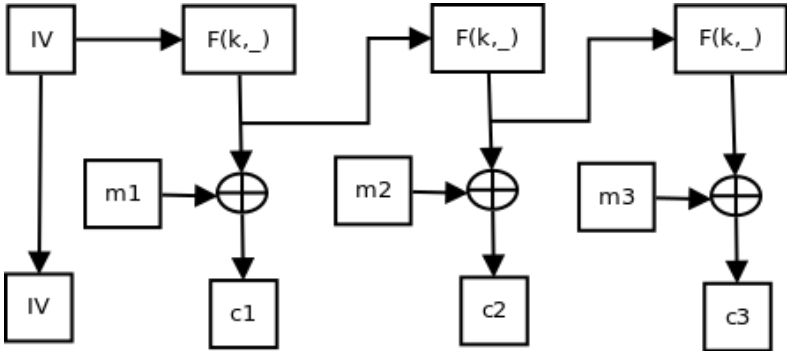


Własności CBC

- Zalety:
  – jeśli F jest permutacją pseudolosową, a wektor początkowy jest losowy, to szyfr jest bezpieczny na atak z wybranym tekstem jawnym
- Wady:
  – przetwarzanie musi być sekwencyjne, nie ma szansy na wykorzystanie ew. równoległości obliczeń
  – błąd w jednym bicie kryptogramu powoduje błąd w dwóch odszyfrowanych blokach

Tryby: OFB

- OFB (*output feedback*)
  – strumień pseudolosowy jest generowany z szyfru blokowego:
  $r_j = Enc(k, r_{j-1}), r_0 = IV$
  – szyfrowanie i odszyfrowywanie: $m_j \oplus r_j$, $c_j \oplus r_j$

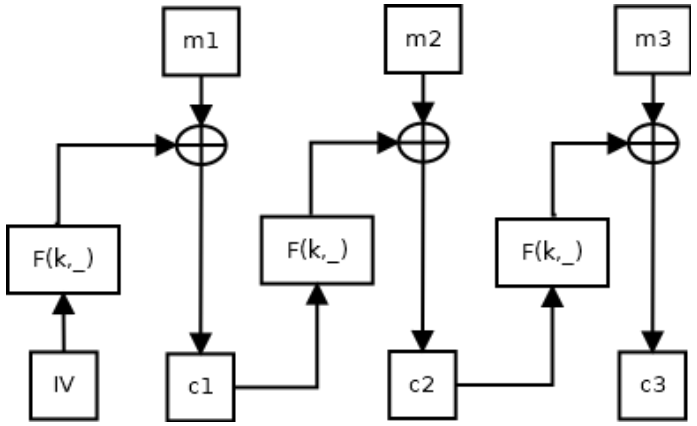


Własności OFB

- Zalety:
  – błędy w szyfrogramie nie propagują się wcale
  – ciąg r_j można przygotować przed otrzymaniem szyfrogramu
  – jeśli F jest funkcją pseudolosową, to szyfr jest bezpieczny na atak z wybranym tekstem jawnym
  – pod warunkiem, że wektor początkowy jest losowy
  – F nie musi być permutacją
- Wady:
  – nie można wykorzystać przetwarzania współbieżnego

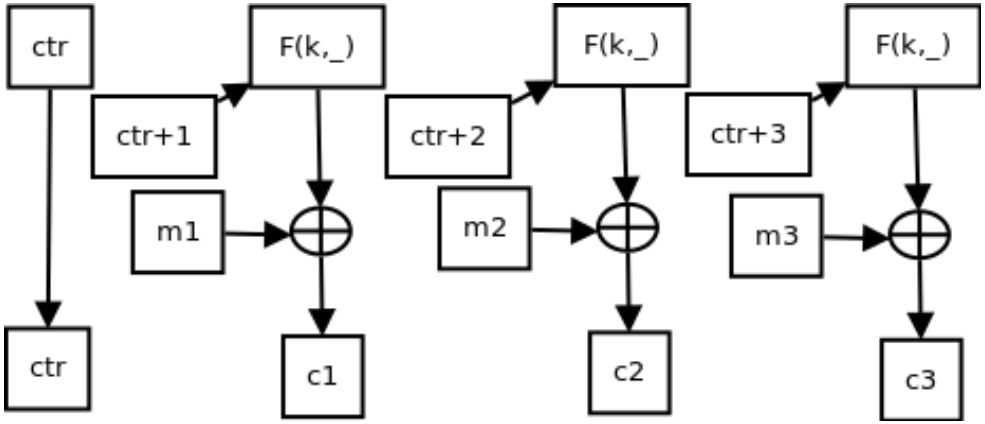
Tryby: CFB

- CFB (*cipher feedback*)
  – $c_j = m_j \oplus Enc(k, c_{j-1})$, $m_j = c_j \oplus Enc(k, c_{j-1})$, $c_0 = IV$
- cechy podobne do CBC, sekwencyjne przetwarzanie, ew. błąd propaguje się na dwa bloki



Tryby: CTR

- CTR (*counter*)
  – strumień pseudolosowy jest generowany z szyfru blokowego: $r_j = Enc(k, ctr + j)$, $ctr = IV$
  – rejestr jest licznikiem, umożliwia działanie współbieżne
  – można szyfrować niezależne fragmenty



Szyfry blokowe

- Bezpieczeństwo zależy od wielkości bloku
  – mniejszy blok zwiększa szansę na powtórzenie wektora losowego
  – $2^{połowa\ długości\ bloku}$ powinno być dużą liczbą
  – blok 64 bitowy dzisiaj jest za mały
- Szyfr blokowy może działać jak szyfr strumieniowy
  – jedyną zaletą szyfru strumieniowego jest wydajność
  – dzisiaj jest to coraz mniej znaczące (może telefony?)

Atak z wybranym kryptogramem

- Ewa ma zgadnąć, która wiadomość jest zaszyfrowana
  – ale ma prawo żądać szyfrowania i odszyfrowania dowolnej wiadomości oprócz dwóch konkursowych
- W trybach OFB, CFB, CTR – kryptogram jest sumą teksty jawnego i pewnego ciągu
  – lekkie zaburzenie powoduje, że wynik jest zbliżony
- W trybie CBC zmiana bloku powoduje zmiany wszystkich dalszych, ale nie wcześniejszych
- Szyfr odporny na atak z wybranym kryptogramem powinien mieć własność odwrotną do odporności na zakłócenia
  – tzn. mała zmiana kryptogramu powoduje dużą zmianę lub niemożność odszyfrowania
- Atak z wybranym kryptogramem jest sens rozpatrywać, gdy przestrzeń tekstów jawnych czy kryptogramów jest ograniczona, może kilobajty, ale nie megabajty długości