Ministerstvo financií
Slovenskej republiky

EFQM
Recognised for excellence
5 star - 2012

Symetrické šifry

M. Stanek

KPMG
cutting through complexity™

Symetrické šifry

Martin Stanek

Obsah

1. Blokové šifry
2. AES, Triple DES
3. Viacnásobné šifrovanie
4. Módy blokových šifier
5. Výplne
6. Prúdové šifry

Blokové šifry – úvod

- Šifrovanie a dešifrovanie – transformácie nad blokom bitov pevnej dĺžky
- E, D: $\{0,1\}^k \times \{0,1\}^n \rightarrow \{0,1\}^n$
- Očákavame korektnosť, teda pre každý kľúč K a každý blok OT P:

$$D_K(E_K(P)) = P$$

- Dĺžka bloku n obvykle 128 bitov (napr. AES) alebo 64 bitov (napr. Triple DES)
- Dlhšie/kratšie správy – módy použitia blokových šifier
- Počet kľúčov 2^k
- Kľúč štandardne volený ako náhodný prvok z $\{0,1\}^k$

Blokové šifry – úvod (2)

- Na dĺžke bloku záleží
  - Krátky blok uľahčuje kryptoanalýzu (hľadanie diferencií a pod.)
  - Veľmi krátky blok: max. $(2^n)!$ permutácií
  - Dlhý blok zvyšuje nároky na HW a implementáciu
- V porovnaní s prúdovými šiframi:
  - Flexibilnejšie (vďaka módom)
  - Častejšie používané

Bezpečnosť

- Úplné preberanie priestoru kľúčov ~ 2^k
- V priemernom prípade (očakávaná zložitosť) ~ 2^{k-1}
- Platí len pre náhodne (uniformne) volené kľúče
- Možné problémy v skutočnosti:
  - Kľúč odvodený z hesla, nekvalitný generátor a pod.
  - Preberanie kľúčov od najpravdepodobnejších

- Čokoľvek lepšie ako úplné preberanie je (teoreticky) úspešný útok – hoci môže byť stále nepraktický
  - Priveľká zložitosť, napr. 2^{118}
  - Nerealistické predpoklady, napr. 2^{90} zvolených OT šifrovaných rovnakým kľúčom

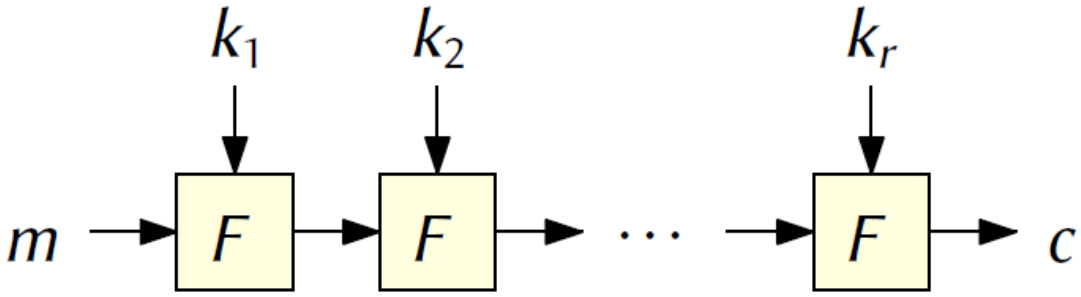
Štandardizácia ISO

- ISO/IEC 18033-3:2010
  - 64-bitový blok: TDEA, MISTY1, CAST-128, HIGHT
  - 128-bitový blok: AES, Camellia, SEED
- ISO/IEC 29192-2:2011 (Lightweight cryptography)
  - 64-bitový blok: PRESENT
  - 128-bitový blok: CLEFIA

- Štandardizované neznamená používané/implementované

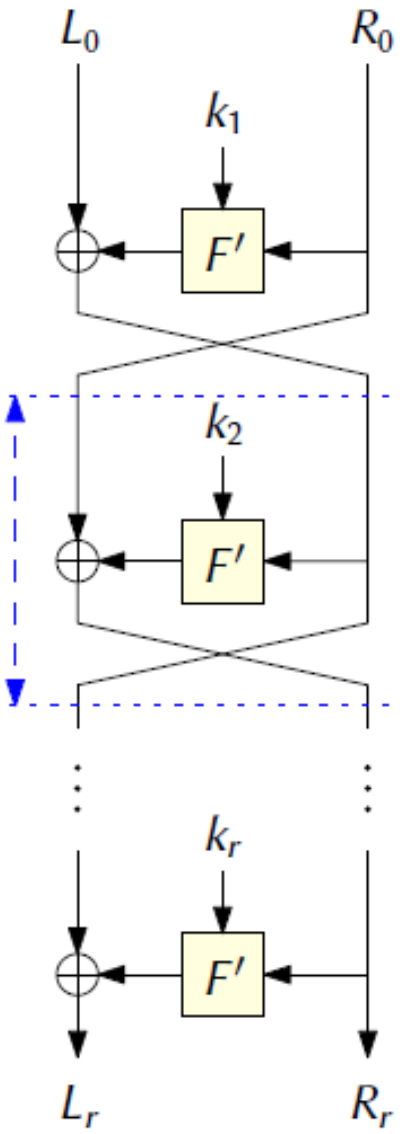
Iterované šifry

- Obvyklá konštrukcia
  - Iterácia jednoduchšej transformácie („kolo“ algoritmu)
  - Plánovanie/expanzia kľúča – vytvorenie podkľúčov pre jednotlivé kolá
  - Zvyčajne nejaká forma „bielenia“ (key whitening)
  - Dešifrovanie využíva inverznú transformáciu F^{-1}
- Napríklad:
  - AES-128 má 10 kôl
  - PRESENT má 32 kôl



Feistelovské šifry

- DES (Triple DES), Camellia, Blowfish, ...
- Konštrukcia kolovej transformácie s „rovnakým“ dešifrovaním
  - Zmena poradia podkľúčov
- Používaná aj v iných konštrukciách – napr.
  - výplňové schémy (OAEP)
  - šifrovanie zachovávajúce formát
- Zovšeobecnenie – nebalancované konštrukcie



AES

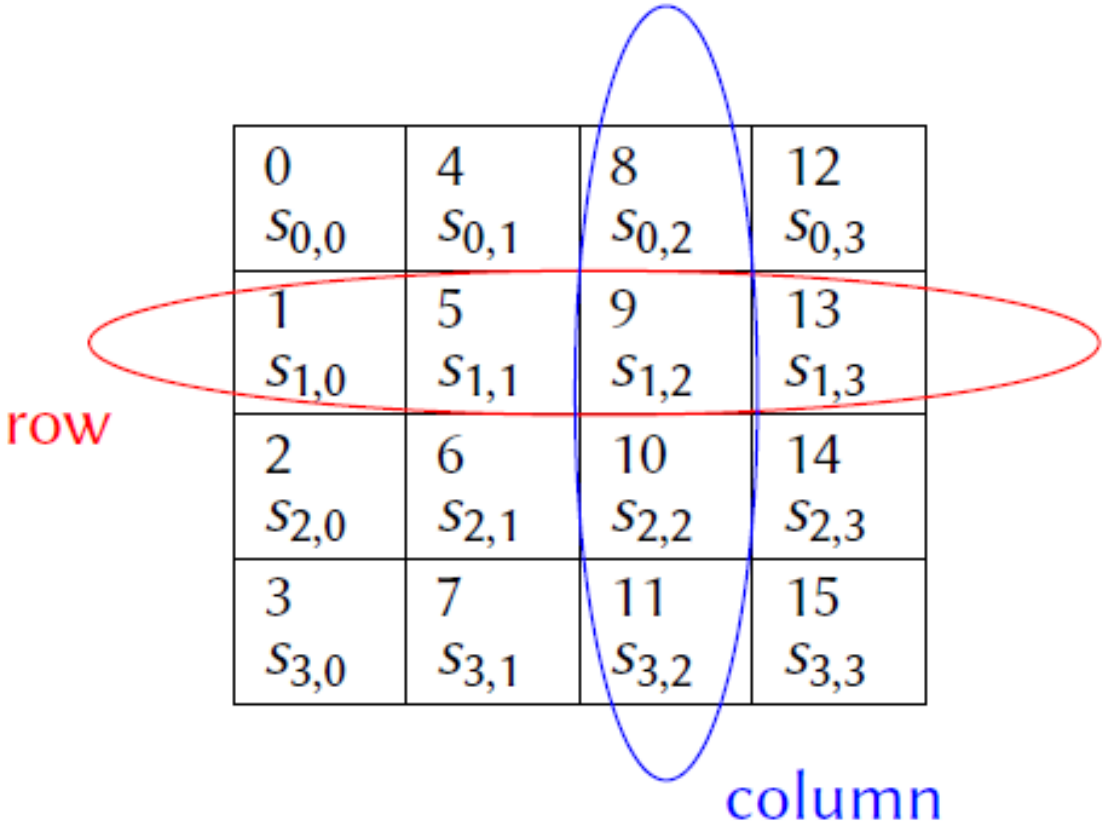
- Algoritmus Rijndael
- AES štandardizovaný 2001 (NIST)
- Najdôležitejšia symetrická šifra súčasnosti
- AES-128 (10 kôl), AES-192 (12 kôl), AES-256 (14 kôl)
  - Zodpovedajúce spomalenie šifrovania a dešifrovania, napr.:

AES-128	42,8 mil. operácií/s
AES-192	36,1 mil. operácií/s
AES-256	31,4 mil. operácií/s

- Dĺžka bloku 128 bitov
- AES nie je Feistelovská šifra

Stav AES

- Otvorený text, vnútorný stav, šifrový text
- Pole 4x4 bajtov



AES operácie

- Kolová transformácia zložená zo 4 operácií:

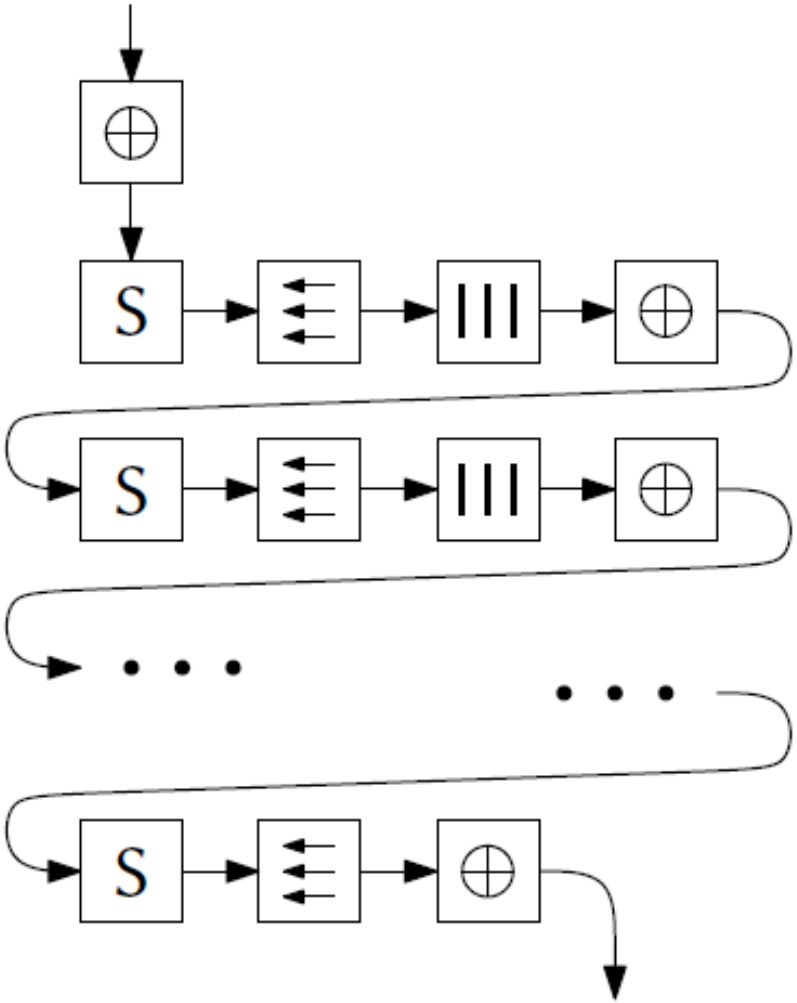
1. Pripočítanie podkľúča (XOR)
2. Substitúcia bajtov
   - Permutácia {0,1,...,255}
3. Posun riadkov
   - Cyklicky doľava (1/2/3/4. riadok o 0/1/2/3 bajtov)
4. Zmiešanie stĺpcov
   - Násobenie stĺpca fixnou maticou

- Všetky operácie sú invertovateľné
- Dešifrovanie: operácie a podkľúče v opačnom poradí

Štruktúra AES

1. kolo

2. kolo



posledné kolo

+ odvodenie podkľúčov

Bezpečnosť AES

- V súčasnosti najlepší útok so získaním kľúča (KPA):

	zložitosť	dáta
AES-128	$2^{126,1}$	2^{88}
AES-192	$2^{189,7}$	2^{80}
AES-256	$2^{254,4}$	2^{40}

Na zamyslenie (1)

Vynechajme v implementácii AES jednu operáciu.

Aké slabiny viete nájsť v AES bez:

- Pripočítania podkľúčov?
- Substitúcie bajtov?
- Posunu riadkov?
- Zmiešania stĺpcov?

Triple DES

- Spolu s AES algoritmus schválený NIST
- Spôsob predĺženia kľúča algoritmu DES (len 56 bitov)
- Viacnásobné šifrovanie (kaskáda)
- Dĺžka bloku 64 bitov
- Tri transformácie DES (E, D) zreťazené za sebou:

$$E^*(P) = E_{K3}(D_{K2}(E_{K1}(P)))$$

$$D^*(C) = D_{K1}(E_{K2}(D_{K3}(C)))$$

- Možnosti voľby kľúčov:

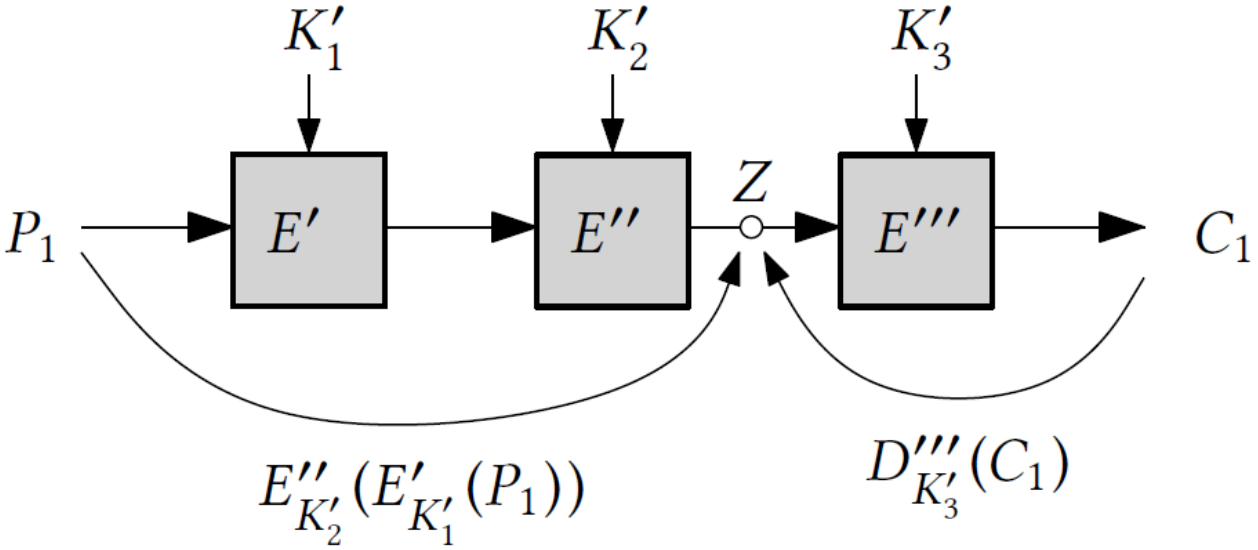
1. Navzájom rôzne K1, K2, K3 (168 bitov, reálna „sila“ 112)
2. K1 = K3, K1 a K2 je rôzne (112 bitov, reálne ~ 80)
3. K1 = K2 = K3 (56 bitov, spätná kompatibilita)

Viacnásobné šifrovanie

- Motivácia:
  - Predĺženie kľúča
  - „Poistka“ pre prípad zlomenia jednej šifry
- Zníženie výkonu šifrovania a dešifrovania
  - Voliť medzi bezpečnosťou a výkonom treba aj pri návrhu jednej (iterovanej) šifry – je 10 kôl pre AES-128 veľa alebo málo?
- Príklad: TrueCrypt
- Bezpečnosť nižšia ako by zodpovedalo dĺžke kľúča
- Niekedy sa bezpečnosť nezvýši
  - Jednoduchá substitucia, permutačná šifra a pod.
- Generický útok: meet-in-the-middle

Meet-in-the-middle útok

- Útok so znalosťou OT (KPA), rôzne typy a dĺžky kaskád
- Príklad: trojité šifrovanie



- Časová zložitosť: 2^{2k} (v prípade Triple DES 2^{112})
- Dvojité šifrovanie: 2^k (bez zvýšenia bezpečnosti)

Vylepšenia MITM

- Napríklad pre Triple DES:

	čas	dáta	
Dva kľúče:	2^{80}	2^{40}	KPA
Tri kľúče:	$2^{108,2}$	2^{45}	CPA

Požiadavky na dáta v KPA/CPA

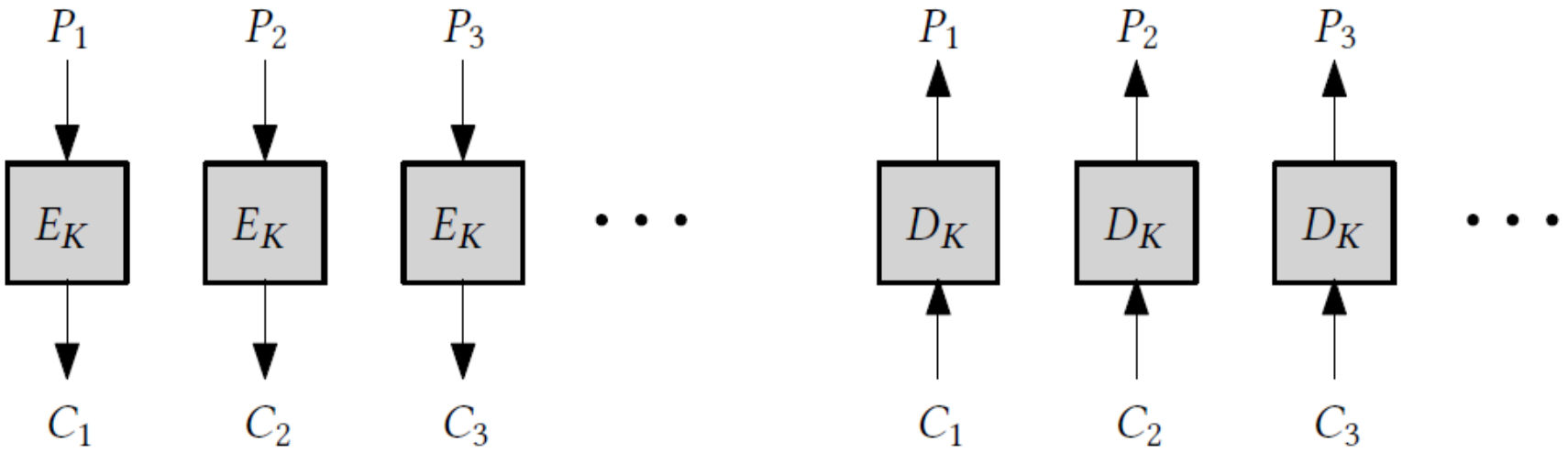
- Predpokladajme 128 bitov dlhý blok
- Veľkosti dát a časy prenosu zohľadňujú len šifrový text

Požiadavka na dáta	Veľkosť v TB	Čas prenosu pri 1 Gb/s
2^{40}	17,6	39 hodín
2^{60}	$1,8 * 10^{8}$	4 676 rokov
2^{80}	$1,9 * 10^{13}$	$4,9 * 10^{9}$ rokov
2^{100}	$2,0 * 10^{19}$	$5,1 * 10^{15}$ rokov

Módy blokových šifier

- Otvorený text obvykle dlhší ako dĺžka bloku
- Rôzne využitie módov
  - Dôvernosť („tradičné“ módy)
  - Autentickosť (bez dôvernosti)
  - Autentizované šifrovanie (autentickosť a dôvernosť)
  - Dôvernosť pre blokové zariadenia
  - „Zabalenie“ kľúča
  - Šifrovanie so zachovaním formátu dát
- Rôzne požiadavky: paralelizovateľnosť, dostupnosť PRNG, rýchlosť, ...

ECB (Electronic Codebook)

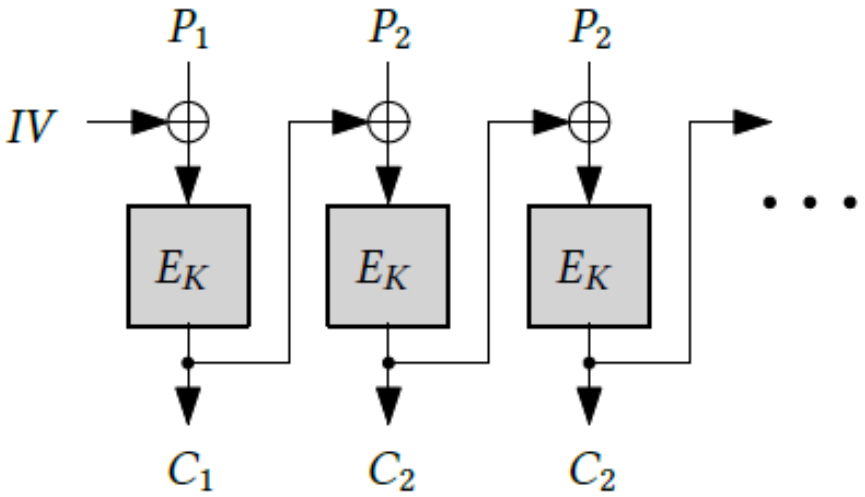


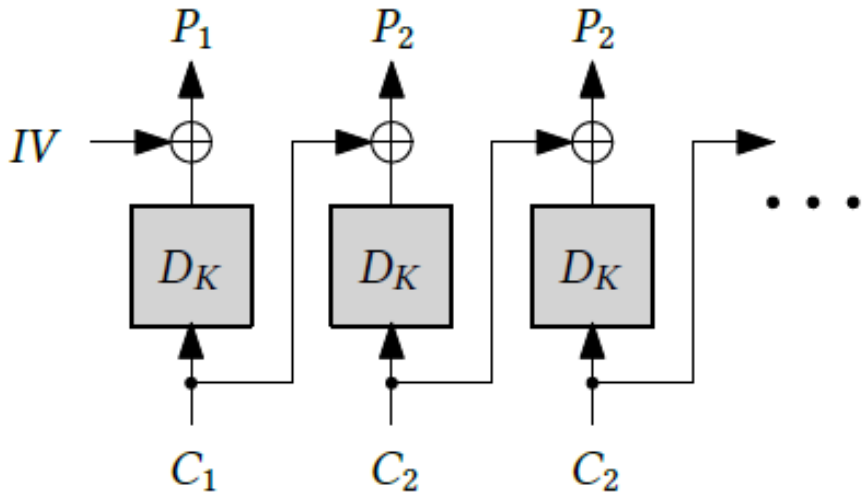
\+ rýchlosť, jednoduchosť, paralelizovateľné, ľahký „seek“

\- zhoda blokov ŠT práve pri zhode blokov OT, ľahká manipulácia s blokmi ŠT (žiadny mód pre dôvernosť nezabezpečí integritu!),

... šifrované heslá Adobe (2013)

CBC (Cipher-block Chaining)





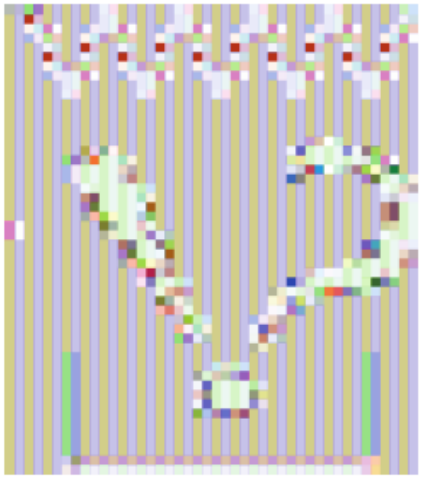
\+ závislosť na IV, menej informácie o OT zo ŠT, paralelizovateľné dešifrovanie

\- sekvenčné šifrovanie, zhoda blokov ŠT vedie k two-time pad problému pre bloky OT (pr. kolízie $2^{-n/2}$, limitovanie dĺžky OT/ŠT)

Ako voliť IV? Náhodne a obvykle posielaný ako C_0.

ECB vs. CBC - vizualizácia

obrázok

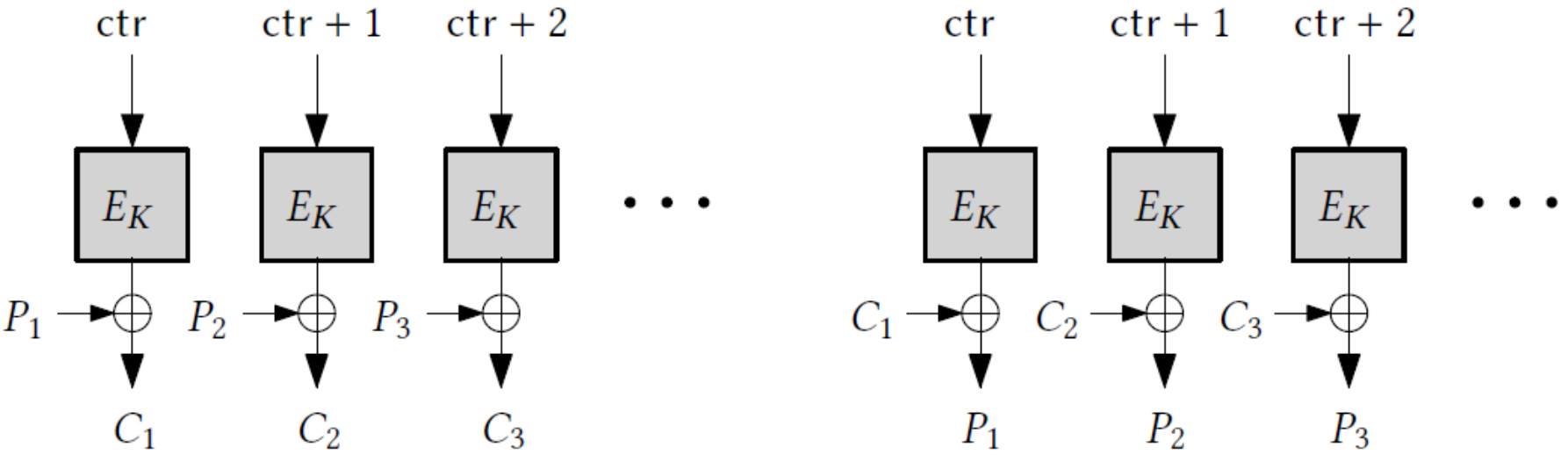


ECB

CBC

AES-128

CTR (Counter)



Prúdová šifra z blokovej šifry, inicializácia ctr z IV

\+ paralelizovateľné, netreba implementovať *D*, ľahký „seek“, garantovaná perióda

\- V prípade „prekrytia“ ctr+i a ctr’+j opäť two-time pad problém (ctr = IV || 0...0, limitovanie dĺžky OT/ŠT)

Na zamyslenie (2)

Ako by ste (čo najjednoduchšie) rozpoznali korektne dešifrovaný text v prirodzenom jazyku, ak skúšate rôzne kľúče pre šifrový text, zašifrovaný napr. AES-128 v ECB móde?

Na zamyslenie (3)

Uvažujme nasledujúci mód (predpokladáme, že dĺžka kľúča a dĺžka bloku je rovnaká):

$$C1 = E_K(P1)$$

$$C2 = E_{P1}(P2)$$

$$C3 = E_{P2}(P3)$$

...

- Ako sa dešifruje?
- Aké vlastnosti má tento mód?
- Má nejaké slabiny?

Výplne (padding)

- Niektoré módy pracujú len s celými blokmi OT/ŠT
- ECB, CBC
- Výplň
  - Pred šifrovaním – zarovnanie OT na násobok dĺžky bloku
  - Po dešifrovaní – získanie pôvodného OT
- Príklad (bajtovej) výplne (PKCS #7, CMS, TLS 1.2):

msg || 01	ak n | |msg| + 1	
msg || 03 03 03	ak n | |msg| + 3	
msg || 10 ... 10	ak n | |msg|	pre n = 128

Padding oracle attack (1)

- Uvažujme CBC mód a nech vieme získať informáciu o korektnej/nekorektnej výplni po dešifrovaní
  - Ako? Chybová hláška, rozdielny čas spracovania a pod.
- Chceme dešifrovať blok ŠT *C* (teda Y = D_K(C))

- Volíme postupne ŠT (predpokladajme 16B blok):

  (X || 00) || C, (X || 01) || C, ..., (X || 7A) || C, ..., (X || FF) || C,
  kde X je reťazec náhodných 15B

  kým nenájdeme ŠT s korektnou výplňou

- Pravdepodobne to znamená, že OT končí bajtom 01

  odtiaľ napr. (7A $\oplus$ Y_{15}) = 01 a teda Y_{15} = 7B

Padding oracle attack (2)

- Pokračujeme ... Aký má byť posledný bajt prvého bloku ŠT, aby sme po dešifrovaní dostali posledný bajt OT 02?

  $(b \oplus Y_{15}) = 02 \Rightarrow b = Y_{15} \oplus 02 = 7B \oplus 02 = 79$

- Volíme postupne ŠT:

  (X || 00 || 79) || C, (X || 01 || 79) || C, ...,

  (X || B2 || 79) || C, ..., (X || FF || 79) || C,

  kde X je reťazec náhodných 14B

  kým nenájdeme ŠT s korektnou výplňou

- To znamená, že OT končí 02 02

  odtiaľ napr. $(B2 \oplus Y_{14}) = 02$ a teda $Y_{14} = B0$

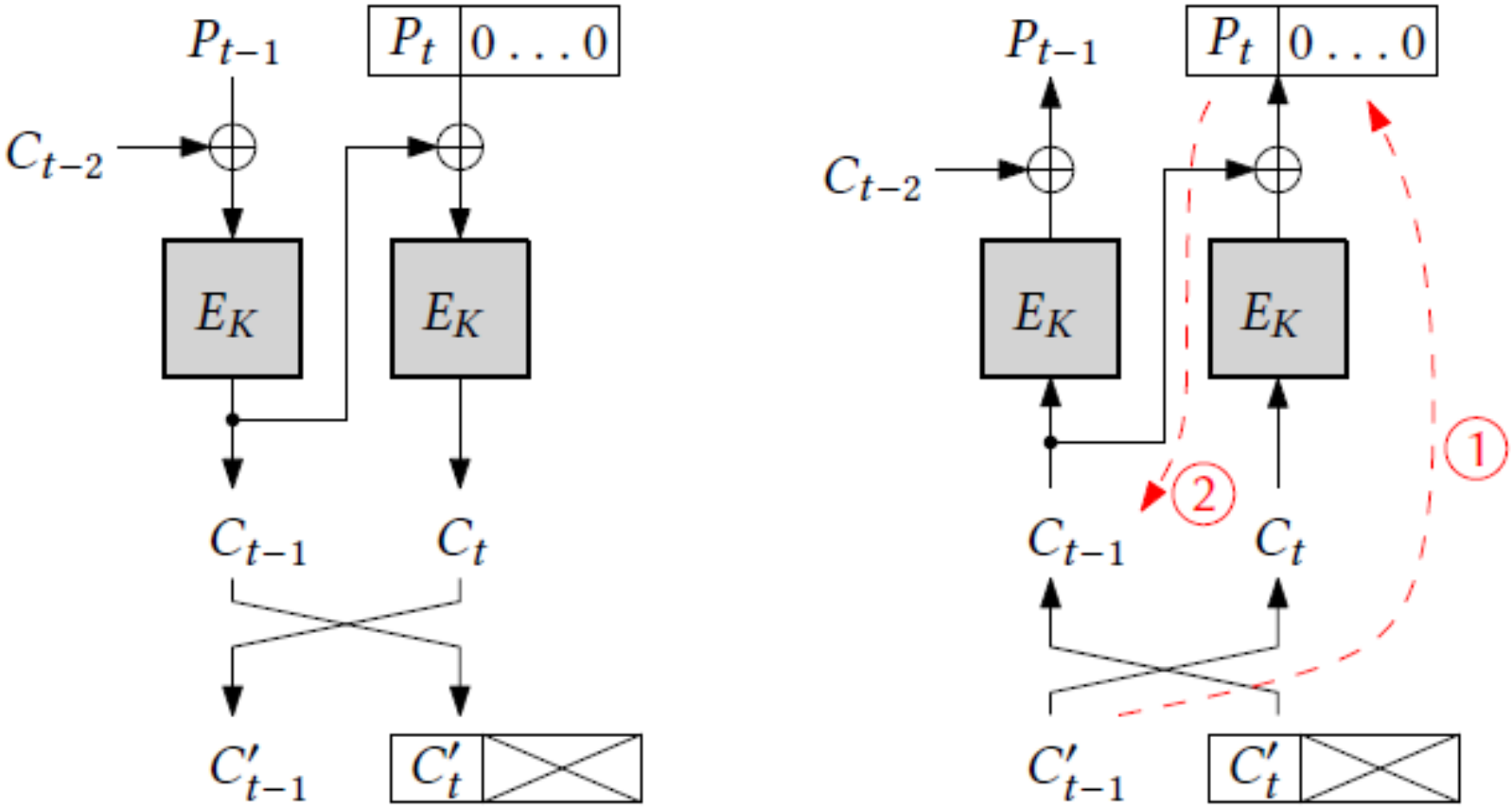
  ... Analogicky postupujeme ďalej

- Variant: útok na SSL/TLS... Lucky Thirteen (2013)

„Kradnutie“ šifrového textu

- Ciphertext stealing
- Spôsob ako sa vyhnúť výplni
- Šifrový text rovnako dlhý ako otvorený text
  - Pre OT s dĺžkou aspoň takou, ako je dĺžka bloku šifry
- ECB a CBC
- Módy pre šifrovanie blokových zariadení (disky)
  - Čo ak sektor nie je násobkom dĺžky bloku šifry?
  - XTS (XEX-based tweaked-codebook mode with ciphertext stealing)

„Kradnutie“ šifrového textu (CBC)



Módy pre autentickosť údajov

- Vytvorenie autentizačného kódu správy
- Nezabezpečujú dôvernosť (!)
- CMAC (Cipher-based MAC)
  - Spracovanie správy podobné CBC + špecifická transformácia posledného bloku
- V praxi sú častejšie používané HMAC konštrukcie (z hašovacích funkcií)

Autentizované šifrovanie

- 1 riešenie pre najčastejšiu kombináciu požiadaviek: dôvernosť + autentickosť
- Obvykle: šifrovanie (dôvernosť) + MAC (autentickosť)
  - 2 prechody cez správu,
  - 2 rôzne algoritmy – ako kombinovať?
- CCM (Counter with CBC-MAC)
  - Kombinácia CTR módu a CBC-MAC konštrukcie
  - Štandardná súčasť 802.11i (WPA2, WiFi Protected Access II)
- GCM (Galois Counter Mode)
  - Kombinácia CTR a „jednoduchého“ spracovania blokov správy
  - TLS, IPSec – voliteľné sady algoritmov
  - IEEE 802.1AE (Media Access Control (MAC) Security)

Šifrovanie blokových zariadení

- Špecifické požiadavky na šifru (mód), napríklad:
  - nemožnosť útočníka podhodením súboru dokázať neskôr jeho prítomnosť na šifrovanom disku
    - CBC mód (kde IV je číslo sektora alebo inak predikovateľný)
    - Úvodné bloky OT P1 a P2 (v rôznych sektoroch) šifrované do rovnakých blokov ak $P1 \oplus IV1 = P2 \oplus IV2$
  - sťažiť nepozorovanú manipuláciu so ŠT
    - Nie je miesto na autentizačný kód
    - CTR a iné „prúdové“ módy nevhodné
- BitLocker – CBC (IV závislý na kľúči) + difúzor
- FileVault 2 – XTS
- TrueCrypt – XTS

XTS

- XEX-based tweaked-codebook mode with ciphertext stealing
- IEEE štandard, schválené NIST

$$C_i = E_{k_1}(P_i \oplus T_n) \oplus T_n; \qquad T_n = E_{k_2}(n) \otimes a^i$$

k_1, k_2 – nezávislé kľúče

n – číslo dátového bloku (napr. 512B)

i – číslo bloku OT v dátovom bloku

a – primitívny prvok v konečnom poli (x)

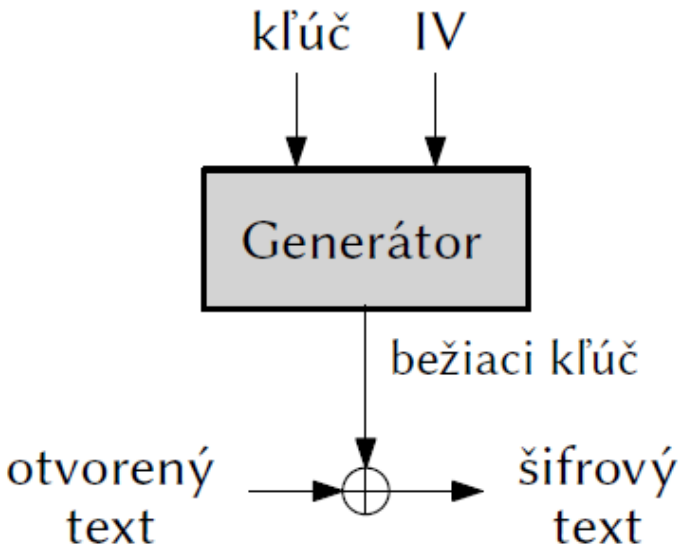
$\otimes$ – násobenie v konečnom poli

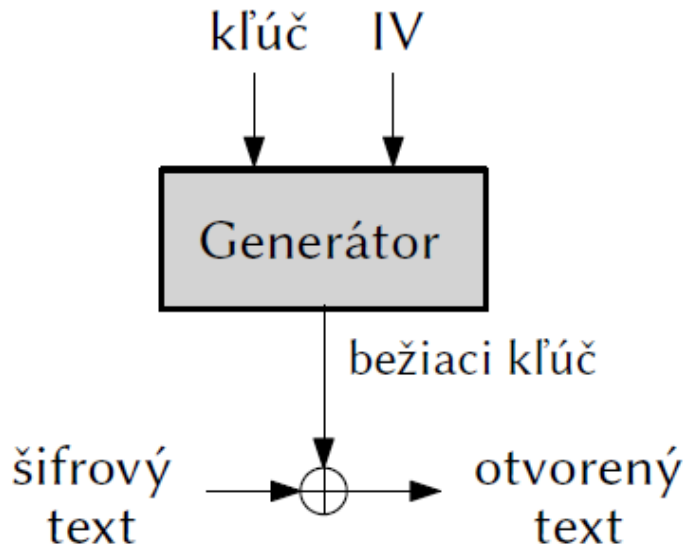
Šifrovanie zachovávajúce formát

- Reťazce pevnej dĺžky nad definovanou abecedou: rodné čísla, čísla platobných kariet, ŠPZ a pod.
- Zachovanie formátu a dĺžky
- NIST SP 800-38 G (draft, 2013)
- FFX schémy
  - založené na Feistelovských štruktúrach

Prúdové šifry

- Deterministický generátor pseudonáhodných čísel
- Inicializácia: kľúč a IV
- Najčastejší variant: aditívna synchrónna šifra
  - Napr. CTR mód, RC4, E0, A5/1, SNOW 3G, ...
- Potenciálne jednoduchšie a rýchlejšie ako blokové šifry





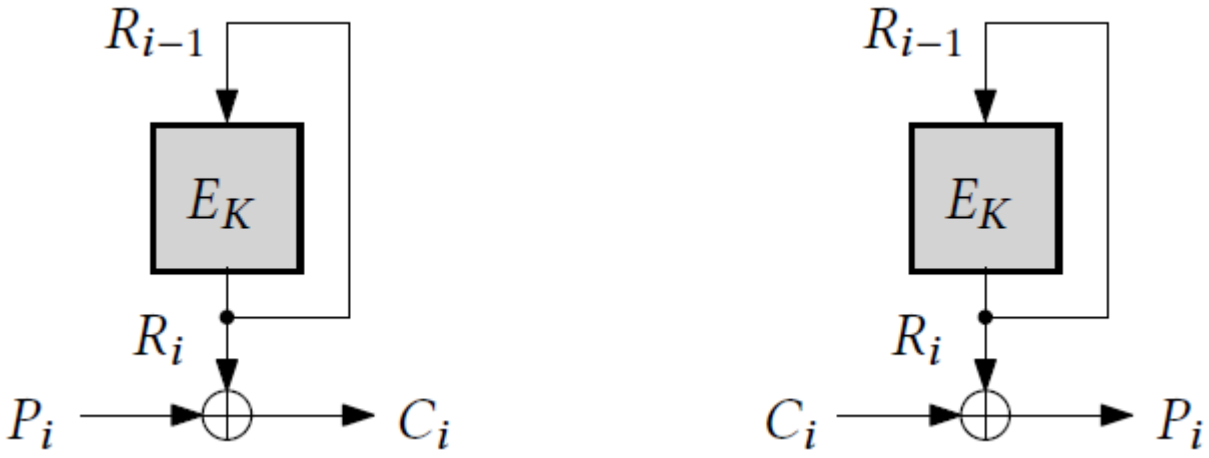
Bezpečnostné požiadavky

- Nutné požiadavky
- Dlhá perióda
- Dobré vlastnosti
  - Prejdenie batériami štatistických testov nestačí
  - Bežiaci kľúč musí byť nepredikovateľný (neodlíšiteľný od náhodnej postupnosti)

Synchrónne prúdové šifry

- Periodické
  - Krátka perióda umožní útok
- Vyžadujú synchronizáciu (stavu) odosielateľa a príjemcu
  - V prípade straty bitu(ov) nezmyselné dešifrovanie
- Ľahké aktívne zasahovanie do ŠT/OT
  - Invertovanie bitov OT
  - Len dôvernosť (ako aj pri štandardných módoch blokových šifier)
- Chyby v ŠT sa nešíria
- V prípade opakovania IV (alebo prekryvu stavov generátora) ... two-time pad problém

OFB (Output Feedback)

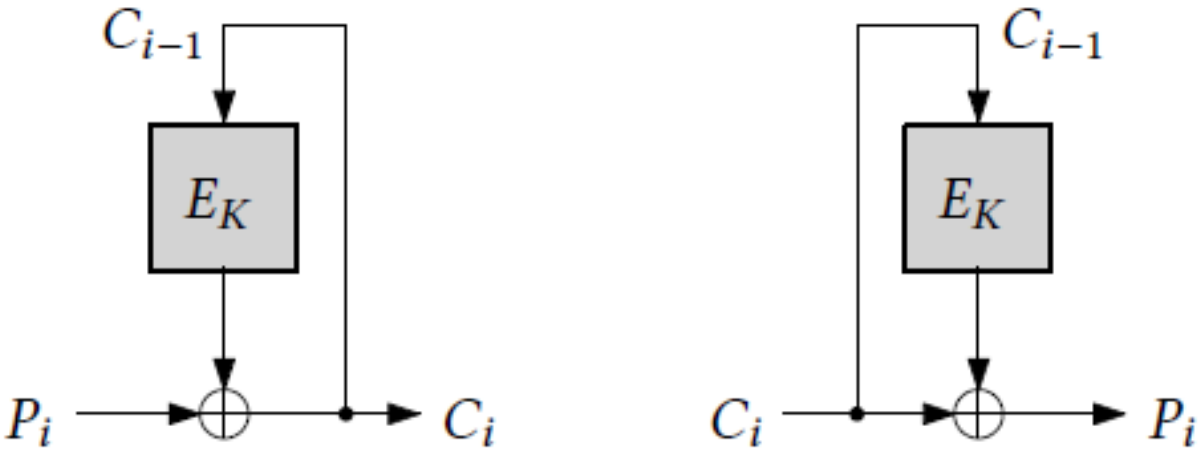


Inicializácia kľúčom a IV (R_0)

\+ netreba implementovať *D*

\- neparalelizovateľné, nemožno spraviť „seek“, pri nedostatkoch v E_K negarantovaná perióda

CFB (Cipher Feedback)



- Samosynchronizujúca prúdová šifra
  - Po strate bloku(ov) sa opäť zosynchronizuje
  - Aperiodická (spätná väzba zo ŠT)
  - Problematické na analýzu

IV je použitý ako C_0

Odlišovací útok

- Cieľ útočníka: odlíšiť bežiaci kľúč od náhodného reťazca
- V KPA scenári útočník ľahko vypočíta bežiaci kľúč
- Význam odlišovacieho útoku:
  - v KPA scenári potvrdiť alebo vyvrátiť použitie konkrétnej prúdovej šifry,
  - v COA scenári testovať rôznych kandidátov pre otvorený text a zvýšiť alebo znížiť pravdepodobnosť, že sú skutočným otvoreným textom,
  - v prípade broadcast prenosu, keď je rovnaký otvorený text posielaný mnohým príjemcom šifrovaný rôznymi kľúčmi, získať čiastočnú informáciu o otvorenom texte.

Príklad – odlišovací útok na RC4

- RC4 generuje bežiaci kľúč ako postupnosť bajtov
- Generátor (*S* je vnútorný stav, inicializovaný ako permutácia, najčastejšie množiny {0,1,...,255}):

$i = 0; j = 0;$
while (is needed):
    $i = (i + 1) \bmod 256;$
    $j = (j + S[i]) \bmod 256;$
    swap$(S[i], S[j])$;
    output $S[(S[i] + S[j]) \bmod 256]$;

- Pre druhý bajt postupnosti platí, že hodnotu 0 nadobúda s pravdepodobnosťou 1/128, namiesto očakávanej 1/256
- Poznámka: V RC4 existujú aj viaceré ďalšie závislosti umožňujúce realizovať odlišovací útok.

Iná slabina v RC4 (Kleinov útok)

- Vo všeobecnosti sú ľubovoľné závislosti a odchýlky od „náhodnosti“ využiteľné na útoky
- V RC4 platí:

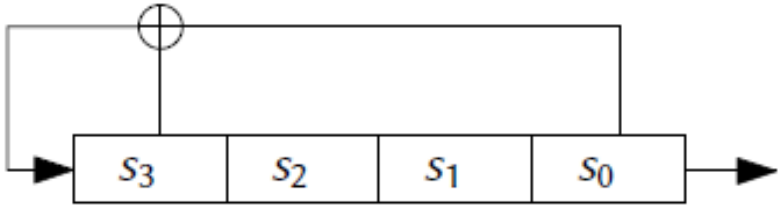
$$\Pr[K[i \bmod k] = S_i^{-1}[i - X[i-1]] - (S_i[i] + j_i)] \approx \frac{1.36}{256}$$

- S_i – je vnútorný stav generátora po *i*-tom kole behu generátora
- X je bežiaci kľúč
- j_i – je hodnota vnútornej premennej generátora po *i*-tom kole.
- pri náhodných a nezávislých hodnotách by uvedený vzťah platil s pravdepodobnosťou 1/256
- rozdiel pravdepodobností a spôsob použitia inicializačného vektora bol základom pre vytvorenie aircrack-ptw na získanie kľúča pre WEP (Wired Equivalent Privacy) šifrovanie

Konštrukčné prvky prúdových šifier

- LFSR – posuvné registre s lineárnou spätnou väzbou
- Často používané
- Ľahké na implementáciu v HW
- Dobré základné štatistické vlastnosti
- Dobrý matematický popis

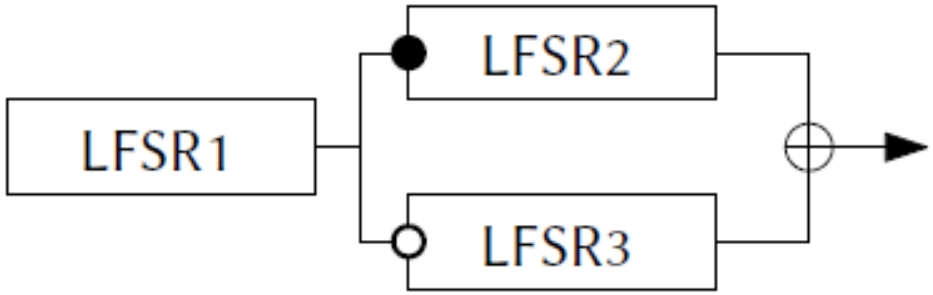
- Nevýhody:
  - Lineárne
  - Ľahko sa dajú syntetizovať



0	0	0	1	0	1	1	0
1	0	0	0	0	0	1	1
1	1	0	0	1	0	0	1
1	1	1	0	0	1	0	0
1	1	1	1	0	0	1	0
0	1	1	1	0	0	0	1
1	0	1	1				
0	1	0	1				
1	0	1	0				
1	1	0	1				

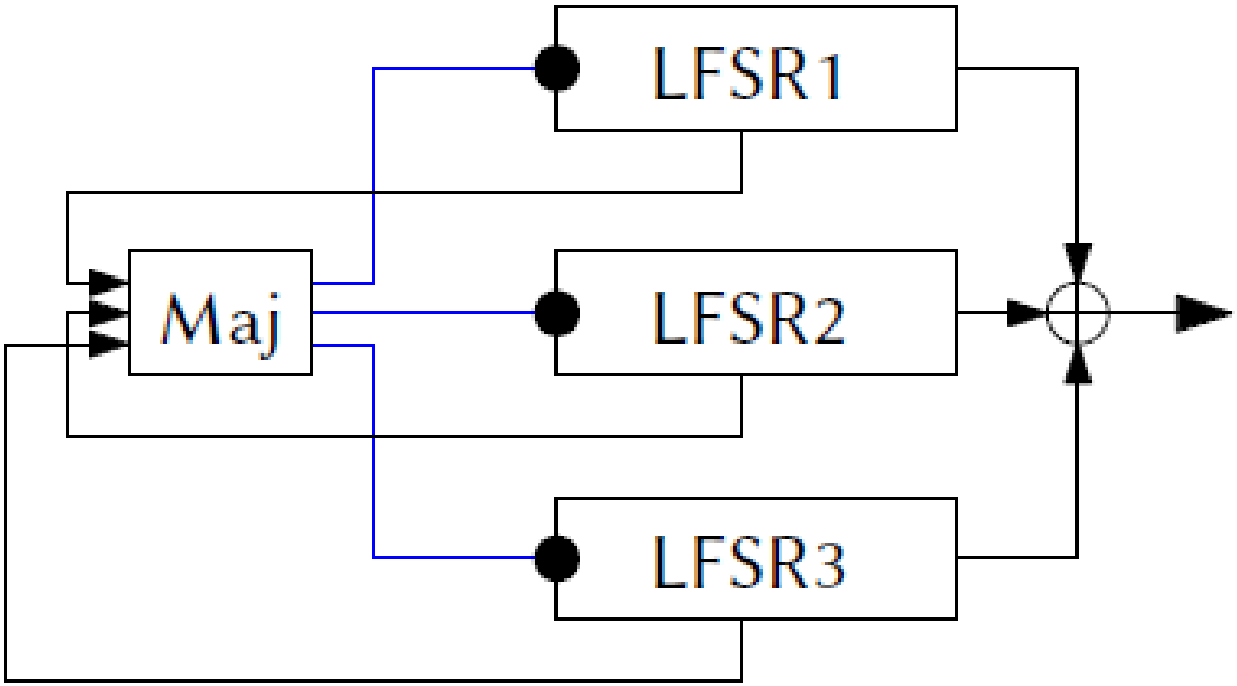
Použitie posuvných registrov

- Filtrované
- Kombinované
- Nepravidelne krokované
- Alternujúce (LFSR1 rozhoduje, ktorý z registrov pôjde, druhý bude stáť)



- Zmršťujúce (LSFR1 rozhoduje, kedy pôjde na výstup generátora výstup z LSFR2)
- A veľa ďalších variantov...

Príklad: A5/1



Na zamyslenie (4)

a) Ako by ste útočili na jednoduchú substitučnú šifru, použitú v OFB móde?

b) Ako by ste útočili na synchrónnu prúdovú šifru s krátkou periódou?

Ďakujem za pozornosť