

# Konstrukce šifer

Andrew Kozlík

KA MFF UK

## Kerckhoffsův princip

- ▶ V roce 1883 stanovil Auguste Kerckhoffs 6 principů, kterými by se měl řídit návrh šifrovacích zařízení.
- ▶ Například, že zařízení by mělo být přenosné, snadno použitelné, kompatibilní s telegrafem apod.  
Nejdůležitější je ale Kerckhoffsův druhý princip:  
*Je třeba, aby šifrovací zařízení nevyžadovalo utajení a aby mohlo padnout do nepřátelských rukou, aniž by to způsobilo potíže.*
- ▶ Jinými slovy: **Bezpečnost šifry nemá spočívat v utajení šifrovacího algoritmu, ale pouze v utajení klíče.**
- ▶ Důvodů je spousta: špionáž, úplatky, vydírání, reverzní inženýrství integrovaných obvodů (mikroskopem) nebo softwaru (z výpisu paměti běžícího programu).

# Budování složitějších šifer

- ▶ **Cíl:** Jedním klíčem (cca 16 bajtů) chceme šifrovat mnoho zpráv (gigabajty).
- ▶ **Problém:** U klasických šifer lze klíč snadno určit z jediného páru otevřeného a šifrového textu.  
Např. u Hillovy šifry vyřešením soustavy lineárních rovnic.
- ▶ **Intuice:** Šifra by měla provázat klíč a otevřený text složitým způsobem tak, aby:
  - ▶ výpočet ŠT z OT a klíče byl rychlý (šifrování),
  - ▶ výpočet OT z ŠT a klíče byl rychlý (dešifrování),
  - ▶ výpočet klíče z OT a ŠT byl složitý (luštění).
- ▶ Požadavky na dobrou šifru jsou ve skutečnosti přísnější ...

# Modely útočníků

## 1. Ciphertext-only attack (COA)

- ▶ Útočník zná jeden nebo více šifrových textů.

## 2. Known-plaintext attack (KPA)

- ▶ Útočník zná jeden nebo více párů ŠT a OT.

## 3. Chosen-plaintext attack (CPA)

- ▶ Útočník volí otevřené texty a dovídá se šifrové texty.

## 4. Chosen-ciphertext attack (CCA)

- ▶ Útočník volí šifrové texty a dovídá se otevřené texty.

- ▶ Někdy se navíc předpokládá, že útočník může zároveň volit i otevřené texty a dovídat se šifrové texty.

- ▶ Všechny šifrové texty vznikají použitím stejného klíče.

# Cíle útočníků

1. Key recovery attack
  - ▶ Určit použitý klíč.
2. Plaintext recovery attack
  - ▶ Dešifrovat nějaký šifrový text.
3. Zjistit jen jeden bit klíče nebo otevřeného textu nebo jiný údaj o klíči nebo otevřeném textu.
4. Distinguishing attack (rozlišovací útok)
  - ▶ Útočník si zvolí dva otevřené texty, ke kterým nezná šifrový text. My náhodně vybereme jeden z nich a zašifrujeme ho. Útočník má z šifrového textu rozlišit, který jsme vybrali, s pravděpodobností úspěchu lepší než 50 %.
  - ▶ Od dobré šifry požadujeme, aby bez znalosti klíče byla při CPA i CCA odolná proti rozlišovacímu útoku.

# Konfuze a difuze

- ▶ Nedostatkem mnoha klasických šifer je, že ŠT odráží statistické charakteristiky OT. Na základě statistické analýzy ŠT pak lze získat informace o klíči.
- ▶ Shannon navrhl dvě metody pro zmaření takové analýzy:
  - ▶ **Difuze:** Rozptyluje statistické charakteristiky OT přes celý ŠT. Čili každý bit OT by měl ovlivňovat co nejvíce bitů ŠT.
  - ▶ **Konfuze:** Každý bit ŠT by měl být složitým způsobem závislý na několika bitech klíče tak, aby vztah mezi ŠT a klíčem byl zastřen složitým systémem nelineárních rovnic.

# Příklady konfuze a difuze

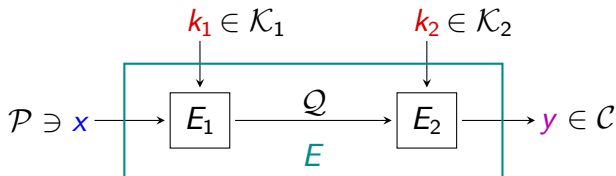
- ▶ **Příklad:** Jednoduchá substituce nebo Vigenèrova šifra.
  - ▶ Neposkytují dobrou difuzi ani konfuzi.
  - ▶ Proto jsou zranitelné frekvenční analýzou.
  
- ▶ **Příklad:** Hillova šifra.
  - ▶ Zajišťuje dobrou difuzi, je-li matice dostatečně velká a hustá. Frekvenční analýza je neúčinná.
  - ▶ Neposkytuje dobrou konfuzi.  
Klíč lze snadno spočítat z OT a ŠT.

# Součin šifer

## Definice

Nechť  $S_1 = (\mathcal{P}, \mathcal{Q}, \mathcal{K}_1, E_1, D_1)$  a  $S_2 = (\mathcal{Q}, \mathcal{C}, \mathcal{K}_2, E_2, D_2)$  jsou šifry. Definujeme *součin šifer*  $S_2 \times S_1 = (\mathcal{P}, \mathcal{C}, (\mathcal{K}_1 \times \mathcal{K}_2), E, D)$ , kde

$$E((k_1, k_2), x) = E_2(k_2, E_1(k_1, x)), \quad \forall x \in \mathcal{P}, \forall (k_1, k_2) \in \mathcal{K}_1 \times \mathcal{K}_2,$$
$$D((k_1, k_2), y) = D_1(k_1, D_2(k_2, y)), \quad \forall y \in \mathcal{C}, \forall (k_1, k_2) \in \mathcal{K}_1 \times \mathcal{K}_2.$$





# Izomorfismus šifer

## Definice

Nechť  $S_1 = (\mathcal{P}, \mathcal{C}, \mathcal{K}_1, E_1, D_1)$  a  $S_2 = (\mathcal{P}, \mathcal{C}, \mathcal{K}_2, E_2, D_2)$  jsou šifry. Říkáme, že  $S_1$  a  $S_2$  jsou *izomorfní*, jestliže existují zobrazení  $f : \mathcal{K}_1 \rightarrow \mathcal{K}_2$  a  $g : \mathcal{K}_2 \rightarrow \mathcal{K}_1$  taková, že

$$E_1(k, x) = E_2(f(k), x), \quad \forall k \in \mathcal{K}_1, \forall x \in \mathcal{P},$$

$$E_2(k, x) = E_1(g(k), x), \quad \forall k \in \mathcal{K}_2, \forall x \in \mathcal{P}.$$

# Idempotentní šifry

## Definice

Jestliže  $S \times S \simeq S$ , říkáme, že  $S$  je *idempotentní šifra*.

- ▶ Jinými slovy, idempotentní šifra je taková, pro kterou platí: Dvojité šifrování dvěma klíči za sebou lze vždy popsat jako jediné šifrování s nějakým jedním klíčem.
- ▶ Čili, vícenásobné šifrování idempotentní šifrou nezvyšuje bezpečnost.
- ▶ Příklady idempotentních šifer:
  - ▶ substituční šifra,
  - ▶ transpoziční šifra,
  - ▶ Hillova šifra,
  - ▶ Vigenèrova šifra,
  - ▶ Vernamova šifra.

# Komutující šifry

## Definice

Jestliže  $S_2 \times S_1 \simeq S_1 \times S_2$ , říkáme, že  $S_1$  a  $S_2$  *komutují*.

- ▶ Jinými slovy, komutující šifry jsou takové, pro které platí: Zašifrování první šifrou a následné zašifrování výstupu druhou šifrou lze vždy popsat jako šifrování v opačném pořadí, byť třeba s jinými klíči.
- ▶ Příklady komutujících šifer:
  - ▶ substituce na písmenech a transpozice na písmenech (klíče budou po změně pořadí stejné,  $f : (k_1, k_2) \mapsto (k_2, k_1)$ ),
  - ▶ transpozice na písmenech a Vigenèrova šifra (klíč Vigenèrovy šifry se může zvětšit).
- ▶ Obecně nekomutují: substituční a Hillova šifra.

# Asociativita součinu šifer

## Pozorování

Operace součinu šifer je asociativní, protože skládání zobrazení je asociativní.

## Pozorování

Jestliže dvě idempotentní šifry  $S_1$  a  $S_2$  komutují, pak  $S_2 \times S_1$  je idempotentní:

$$(S_2 \times S_1) \times (S_2 \times S_1) \simeq S_2 \times S_2 \times S_1 \times S_1 \simeq S_2 \times S_1.$$

## Příklad

Označme  $S$  substituční šifru na písmenech a  $T$  transpoziční šifru na písmenech, pak

$$S_{k_6} \circ T_{k_5} \circ S_{k_4} \circ T_{k_3} \circ S_{k_2} \circ T_{k_1} \simeq S_{k''} \circ T_{k'}.$$

# Budování složitějších šifer

- ▶ Složitější šifru můžeme vybudovat z neidempotentní šifry  $S$  iterováním:  $S^n = S \times S \times \cdots \times S$ .
- ▶ Jednu iteraci šifry  $S$  nazýváme *runda*.
- ▶ Kde vzít neidempotentní šifru?
- ▶ Můžeme vzít dvě šifry, které nekomutují.  
Jejich součin by obecně neměl být idempotentní.

# Nelineární operace

- ▶ K zajištění nelinearity se v mnoha šifrách používá tzv. *substituční box* (S-box).
- ▶ S-box je zobrazení  $S : \{0, 1\}^m \rightarrow \{0, 1\}^n$ .
- ▶ Obvykle  $m = n$  a  $S$  je bijekce, čili substituční šifra.
- ▶ Některé šifry používají S-boxy závislé na klíči (např. Twofish).
- ▶ Většina šifer používá jeden nebo více konstantních S-boxů (např. DES a AES) a klíč se aplikuje před použitím S-boxu.

## S-boxy

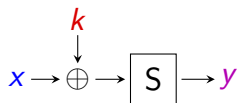
- ▶ S-boxy se zpravidla implementují vyhledávací tabulkou, protože nemívají jednoduchý algebraický popis.

Vstup	Výstup
000000	1110
000001	0000
000010	0100
000011	1111
000100	1101
⋮	⋮
111110	0000
111111	1101

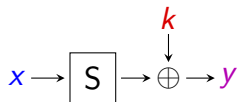
- ▶ Velikost S-boxu roste exponenciálně v počtu vstupních bitů  $m$ , proto většinou  $m \leq 8$ .

# Využití konstantního bijektivního S-boxu

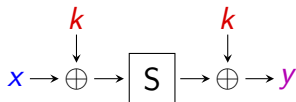
- ▶ Pokusíme se posílit Vernamovu šifru proti KPA.
- ▶ Přidání S-boxu na konec nemá pro bezpečnost význam:



- ▶ Přidání S-boxu na začátek ztíží COA tím, že homogenizuje entropii bitů OT, ale KPA zůstává triviální:

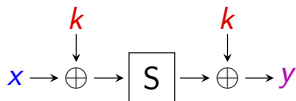


- ▶ V následujícím uspořádání (tzv. *Evenovo-Mansourovo schéma*) už obecně nelze z  $x$  a  $y$  vypočítat  $k$  tak snadno:





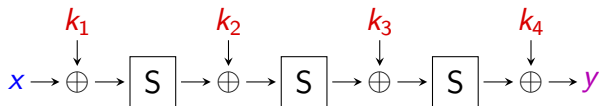
# Evenovo-Mansourovo schéma



- ▶ Dobrý S-box zde tvoří překážku, přes kterou lze „přenést“ známou hodnotu snadno, ale neznámou hodnotu nikoliv.
- ▶ Evenovo-Mansourovo schéma pak poskytuje dobrou konfuzi.
- ▶ **Problém:** Můžeme si dovolit jen malý S-box ( $m \approx 8$  bitů).
  - ▶ Toto schéma tak samo o sobě neposkytuje dobrou difuzi.
  - ▶ Klíč o délce  $m$  bitů odhalíme hrubou silou.

# S-box nestačí

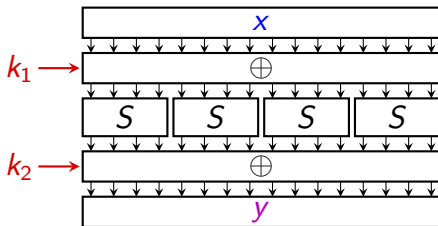
- ▶ Klíč by měl mít přes 80 bitů, aby mohl odolat útoku hrubou silou.
- ▶ **Řešení:** Přidáme víc klíče,  $k = (k_1, k_2, k_3, k_4)$ :



- ▶ **Problém:** Schéma dovoluje jen malou délku OT a ŠT.
  - ▶ Pro pevně zvolený klíč si útočník může při KPA vytvořit slovník dvojic  $(x, E_k(x))$ , který pokryje většinu  $x \in \mathcal{P}$ .
  - ▶ Množina  $\mathcal{P}$  musí být dostatečně velká (délka OT alespoň 64 bitů), aby toto nebylo snadno proveditelné.

## S-box nestačí

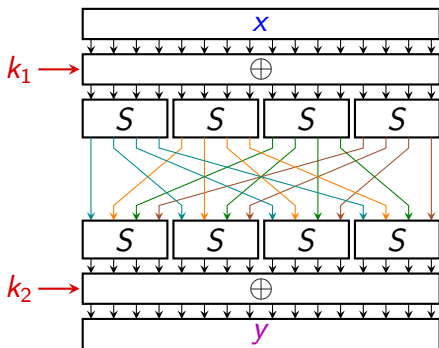
- ▶ Další možností je rozmístění S-boxů paralelně:



- ▶ Máme vedle sebe několik Evenových-Mansourových schémat, která spolu nijak neinteragují.
- ▶ Toto řešení není o moc lepší, protože složitost útoků vzrůstá přímo úměrně k počtu S-boxů, resp. k délce OT nebo klíče.
- ▶ Složitost útoků by měla vzrůstat exponenciálně vzhledem k délce klíče.

# Difuzní vrstva

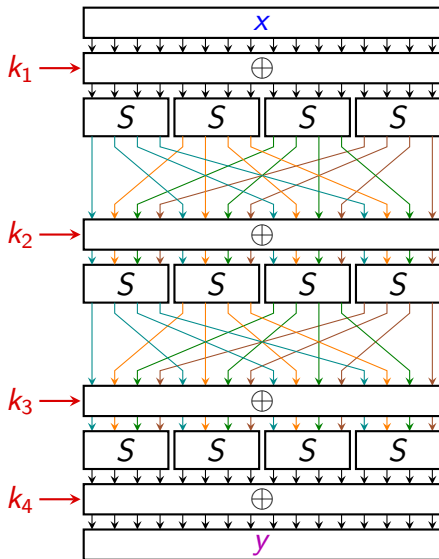
- ▶ Dobrým řešením je provázat několik paralelních konfuzních operací pomocí jedné široké difuzní operace.
- ▶ Jednoduchou difuzní operací je např. bitová permutace:



- ▶ Lepší difuze lze docílit použitím lineárního zobrazení, jehož matice je MDS (maximum distance separable).

# Substituční-permutační šifra

- Konfuzi lze dále vylepšit iterováním tohoto schématu.



# MDS matice

## Definice

Nechť  $\mathbf{A}$  je matice typu  $m \times n$  nad konečným tělesem  $\mathbb{F}_q$ .  
Jestliže  $|\mathbf{Ax}| + |\mathbf{x}| > m$  pro všechna  $\mathbf{x} \in \mathbb{F}_q^n \setminus \mathbf{0}$ , pak říkáme, že  $\mathbf{A}$  je *MDS matice* (maximum distance separable).

- ▶  $|\mathbf{x}|$  značí Hammingovu váhu vektoru  $\mathbf{x}$ , tj. počet nenulových složek v  $\mathbf{x}$ .
- ▶ Jinými slovy, je-li  $\mathbf{A}$  MDS matice, pak platí, že změníme-li  $t$  složek vektoru  $\mathbf{x}$ , pak se změní alespoň  $m - t + 1$  složek vektoru  $\mathbf{Ax}$ .
- ▶ Změna jedné složky v  $\mathbf{x}$  způsobí změnu všech složek v  $\mathbf{Ax}$ .
- ▶ Změna dvou složek v  $\mathbf{x}$  způsobí změnu v alespoň  $m - 1$  složkách  $\mathbf{Ax}$ , atd.