

Data Encryption Standard (DES)

Andrew Kozlík

KA MFF UK

Šifra DES

- ▶ DES je bloková šifra, $\mathcal{P} = \mathcal{C} = \{0, 1\}^{64}$.

- ▶ Klíče mají délku 64 bitů, ale jen 56 bitů je účinných:

$$\mathcal{K} = \{ \mathbf{b} \in \{0, 1\}^{64} \mid \sum_{i=1}^8 b_{i+8n} \equiv 1 \pmod{2}, 0 \leq n \leq 7 \},$$

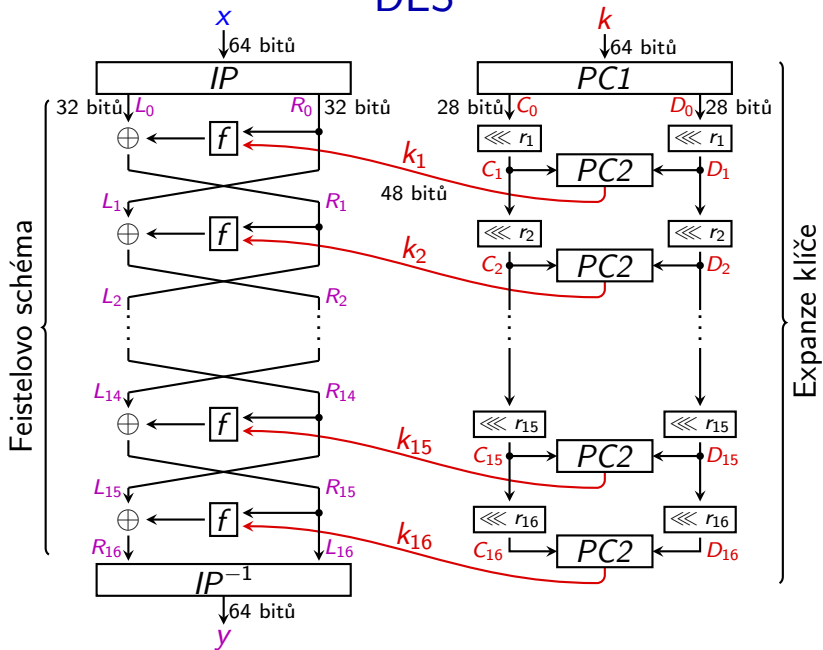
kde $\mathbf{b} = (b_1, \dots, b_{64})$.

- ▶ Čili rozdělíme-li klíč na 8 bajtů, pak každý má lichou paritu.
- ▶ Šifra ignoruje bity klíče $b_8, b_{16}, b_{24}, b_{32}, b_{40}, b_{48}, b_{56}$ a b_{64} .
- ▶ DES byl navržen pro hardwarovou implementaci.

Historie DESu

- ▶ Standard publikoval americký National Institute of Standards and Technology v roce 1977.
- ▶ Původní návrh z roku 1974 vycházel z šifry LUCIFER, kterou vyvinul Horst Feistel z IBM.
- ▶ NSA si vynutila zkrácení klíče z navrhovaných 64 bitů na účinných 56 bitů. Původně NSA usilovala o 48 bitů.
- ▶ NSA vyměnila S-boxy, čímž učinila šifru odolnou proti diferenciální kryptoanalýze.
- ▶ Diferenciální kryptoanalýza byla objevena Bihamem a Shamirem v r. 1990, ale NSA ji znala již při designu.
- ▶ Dlouho panovaly obavy, že změnou S-boxů si NSA vytvořila zadní vrátka. Ve skutečnosti ale šifru vylepšila.

DES



Šifra DES

- ▶ Na otevřený text se nejdříve aplikuje bitová permutace IP .
- ▶ IP (initial permutation):
 - ▶ Usnadňuje hardwarovou implementaci při načítání OT přes 8bitovou sběrnici.
 - ▶ Nemá žádný význam pro bezpečnost šifry.
- ▶ Výsledek permutace se rozpůlí na dvě 32bitové části L_0 a R_0 , které projdou 16 rundami Feistelova schématu.
- ▶ Na závěr se R_{16} a L_{16} spojí a provede se inverzní bitová permutace IP^{-1} .

Feistelovo schéma

- ▶ Pro $i = 1, \dots, 16$:

$$L_i = R_{i-1}$$

$$R_i = L_{i-1} \oplus f(R_{i-1}, k_i)$$

- ▶ Každá runda je invertovatelná bez ohledu na definici f :

$$R_{i-1} = L_i$$

$$L_{i-1} = R_i \oplus f(R_{i-1}, k_i) = R_i \oplus f(L_i, k_i)$$

- ▶ V poslední rundě nedochází k záměně levé a pravé poloviny. Výstupem schématu je (R_{16}, L_{16}) .

- ▶ **Důsledek:** Dešifrování se provádí shodně jako šifrování, pouze s rozdílem, že se obrátí pořadí rundovních klíčů.

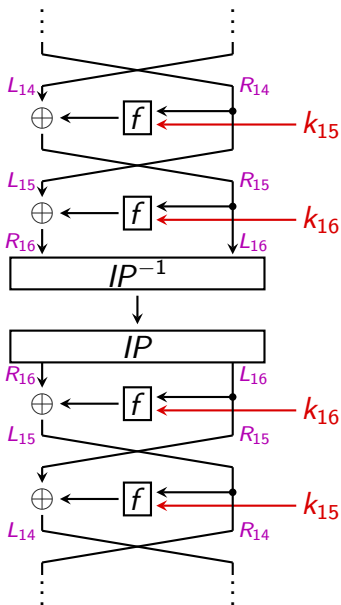
Důkaz vzorci:

- ▶ Vzorci pro dešifrování rundy jsou stejné jako pro šifrování, mají jen obráceně pojmenované proměnné $L \leftrightarrow R$.
- ▶ Dáme-li na vstup šifry DES šifrový text, tak do Feistelova schématu vstupuje (R_{16}, L_{16}) , tj. proměnné jsou obráceně pojmenované.

Dešifrování

Důkaz obrázkem:

Zřetězíme dvě Feistelova schémata, kde druhé má rundovní klíče v opačném pořadí než první. Vidíme, že jednotlivé operace se vzájemně vyloučí.



S-boxy

- ▶ Šifra DES má 8 konstantních S-boxů $\{0, 1\}^6 \rightarrow \{0, 1\}^4$.

- ▶ Například S_1 vypadá takto:

S_1	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	14	4	13	1	2	15	11	8	3	10	6	12	5	9	0	7
1	0	15	7	4	14	2	13	1	10	6	12	11	9	5	3	8
2	4	1	14	8	13	6	2	11	15	12	9	7	3	10	5	0
3	15	12	8	2	4	9	1	7	5	11	3	14	10	0	6	13

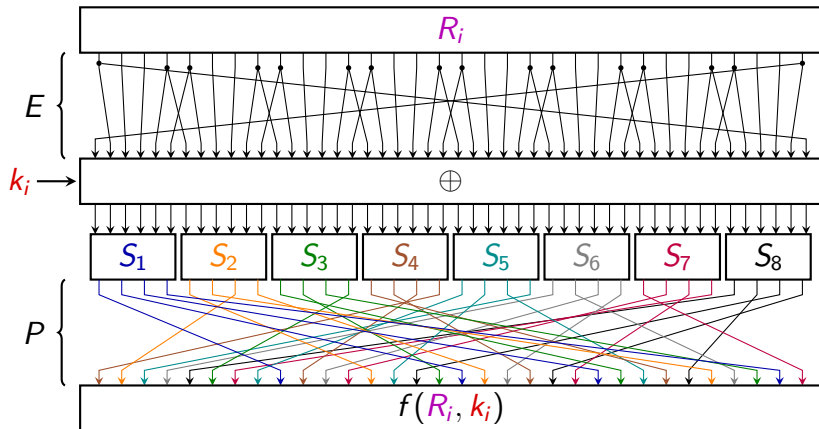
- ▶ Interpretace: První a šestý bit vstupu určují řádek a prostřední čtyři bity vstupu určují sloupec, kde leží výsledek.
- ▶ Příklad: $S_1(101010) = 0110$.
 $101010 \rightarrow$ řádek 2 a sloupec 5 $\rightarrow 6 = (0110)_2$.
- ▶ Řádky S-boxů jsou permutace hodnot $\{0, \dots, 15\}$.
- ▶ S-boxy jsou jediným nelineárním prvkem DESu a jádrem jeho bezpečnosti.

Rundovní funkce f

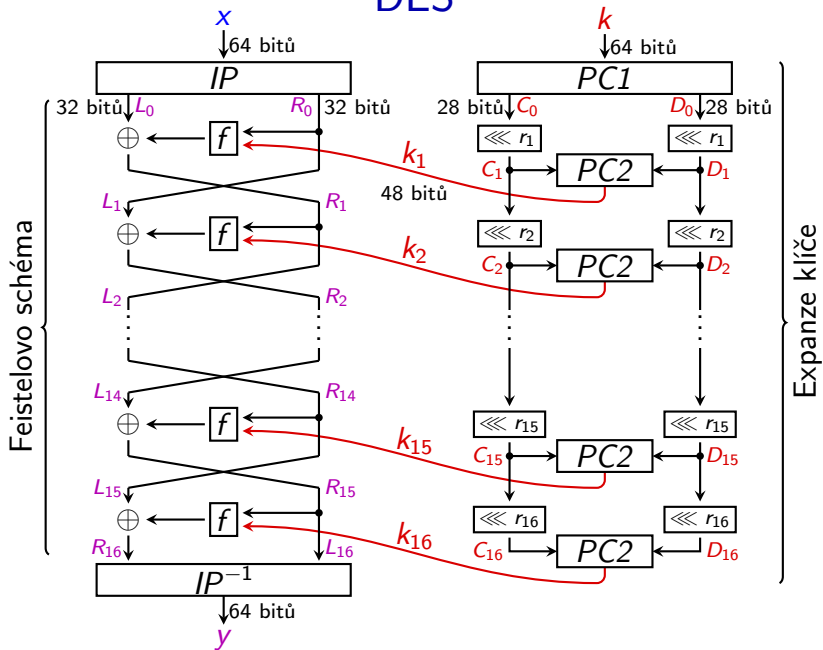
- ▶ Funkce f sestává z posloupnosti čtyř operací:
- ▶ E : Bitová expanze, která duplikuje polovinu bitů (32 bitů \rightarrow 48 bitů).
- ▶ Přičtení 48bitového rundovního klíče operací XOR.
- ▶ S_1, \dots, S_8 : Vrstva osmi S-boxů (48 bitů \rightarrow 32 bitů).
- ▶ P : Bitová permutace na 32 bitech.
Permutace zajišťuje, že výstupní bity každého S-boxu putují v další rundě do šesti jiných S-boxů.

Rundovní funkce f

► $f(R_i, k_i) = P(S(k_i \oplus E(R_i)))$

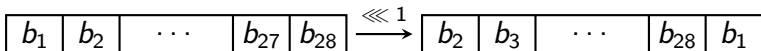


DES



Expanze klíče

- ▶ **Anglicky:** key schedule.
- ▶ Expanze klíče vytvoří z hlavního 64bitového klíče šestnáct 48bitových rundovních klíčů.
- ▶ Permutace *PC1* (permuted choice)
 - ▶ Zahodí poslední bit každého bajtu.
 - ▶ Permutuje zbylé bity klíče. (Nemá význam pro bezpečnost.)
 - ▶ Rozdělí je na dvě poloviny o velikosti 28 bitů.
- ▶ Permutace *PC2*
 - ▶ Z 56 vstupních bitů vybere 48 a permutuje je.
- ▶ Permutace „ $\lll r$ “ rotuje bity doleva o r pozic.



Expanze klíče

- ▶ Při expanzi klíče dochází pouze k permutacím a výběru bitů hlavního klíče. Každý rundovní klíč je pouze výběrem určitých bitů hlavního klíče.
- ▶ Každý bit hlavního klíče se dostane do přibližně 14 rundovních klíčů z celkových 16.

- ▶ Registry C a D se v průběhu expanze zcela protočí:

i	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
r_i	1	1	2	2	2	2	2	2	1	2	2	2	2	2	2	1

$\sum_{i=1}^{16} r_i = 28$, čili $C_{16} = C_0$ a $D_{16} = D_0$.

- ▶ Šifrování dalšího bloku může přímo pokračovat, aniž by bylo třeba reinitializovat algoritmus expanze klíče.

Dešifrování a expanze klíče

- ▶ Dešifrování se provádí krmením rundovních klíčů do Feistelova schématu v obráceném pořadí.
- ▶ K tomuto stačí spustit expanzi klíče ve zpětném chodu.
- ▶ Dešifrování lze tedy provést tak, že pouze nahradíme v expanzi klíče operaci $\lll r_i$ operací $\ggg r_{17-i}$.

Bezpečnost

- ▶ Všechny komponenty šifry kromě S-boxů jsou lineární ($g(x) \oplus g(y) = g(x \oplus y)$).
- ▶ Kdyby i S-boxy byly lineární, pak by šifra šla popsat 64 lineárními rovnicemi o $64 + 56$ proměnných.
- ▶ Ukazuje se, že není na újmu bezpečnosti, že S-boxy jsou jedinou nelineární komponentou.
- ▶ Útok hrubou silou:
Už v době návrhu existovaly obavy, že klíč je příliš krátký.
 - ▶ 1977 Diffie a Hellman navrhly stroj v ceně \$20 000 000, který by provedl útok za 1 den.
 - ▶ 1993 Wiener navrhl stroj v ceně \$1 000 000, který by provedl útok za 7 hodin.
 - ▶ 1998 Electronic Frontier Foundation sestavila stroj za \$250 000, který provedl útok v 56 hodinách.

Bezpečnost

- ▶ Diferenciální kryptoanalýza (chosen plaintext attack):
 - ▶ Vyžaduje 2^{49} volených otevřených textů a příslušných šifrových textů zašifrovaných stejným klíčem.
 - ▶ Nepraktické.

- ▶ Lineární kryptoanalýza (known plaintext attack):
 - ▶ Objevena v roce 1992 Mitsuruem Matsumim.
 - ▶ Vyžaduje 2^{43} párů otevřených a šifrových textů.
 - ▶ Nepraktické.
 - ▶ 1994 První povedený útok na DES. (12 CPU, 40 dnů se generovaly páry OT a ŠT, 10 dnů trvala kryptoanalýza.)

Vlastnosti DESu

- ▶ Bitová negace je operace, která obrací hodnotu každého bitu. Značíme ji pruhem: $\overline{1011} = 0100$.
- ▶ Pro libovolné $x \in \mathcal{P}$ a $k \in \mathcal{K}$ platí $\text{DES}(\overline{k}, \overline{x}) = \overline{\text{DES}(k, x)}$.
 - ▶ Negace klíče způsobí negace rundovních klíčů (ty se skládají z bitů hlavního klíče).
 - ▶ Negace otevřeného textu způsobí negaci vstupu do obou větví Feistelova schématu.
 - ▶ V rundovní funkci f se negace R_i a k_i pokaždé vyruší:

$$\begin{aligned}f(\overline{R}_i, \overline{k}_i) &= P(S(\overline{k}_i \oplus E(\overline{R}_i))) \\ &= P(S(\overline{k}_i \oplus \overline{E(R_i)})) \\ &= P(S(k_i \oplus E(R_i))) = f(R_i, k_i).\end{aligned}$$

- ▶ Negace vstupu se tedy šíří celým Feistelovým schématem:

$$\overline{L}_i \oplus f(\overline{R}_i, \overline{k}_i) = \overline{L}_i \oplus f(R_i, k_i) = \overline{L_i \oplus f(R_i, k_i)} = \overline{R_{i+1}}.$$

Útok hrubou silou

- ▶ Známe jeden pár OT a ŠT a provádíme útok hrubou silou. Kolik volání šifrovací funkce je třeba v průměru provést?
- ▶ Předpokládáme, že klíč byl zvolen náhodně s rovnoměrným rozdělením.
- ▶ Postupně zkusíme jednotlivé klíče $k \in \mathcal{K}$ a ověříme, zda $y \stackrel{?}{=} E(k, x)$.
- ▶ Pravděpodobnost, že k nalezení správného klíče bude třeba právě i pokusů je $\frac{1}{|\mathcal{K}|}$.
- ▶ Střední hodnota počtu pokusů je tedy

$$\sum_{i=1}^{|\mathcal{K}|} \frac{1}{|\mathcal{K}|} \cdot i = \frac{1}{|\mathcal{K}|} \frac{|\mathcal{K}|(|\mathcal{K}| + 1)}{2} = \frac{|\mathcal{K}| + 1}{2}.$$

- ▶ V případě DESu tedy 2^{55} volání šifrovací funkce.

Cvičení

Známe x , y_1 a y_2 takové, že

$$y_1 = \text{DES}(k, x) \quad \text{a} \quad y_2 = \text{DES}(k, \bar{x}).$$

Popište útok, který odhalí klíč k s průměrnou časovou složitostí 2^{54} šifrovacích operací DES.

Meet-in-the-middle útok

- ▶ Known plaintext attack na součinné šifry $S_2 \times S_1$.
- ▶ Známe x a y takové, že $y = E_2(k_2, E_1(k_1, x))$ a chceme odhalit k_1 a k_2 .
- ▶ Pro každý klíč $k'_1 \in \mathcal{K}_1$ spočteme $E_1(k'_1, x)$ a sestrojíme vyhledávací tabulku,

$E_1(k'_1, x)$	k'_1
\vdots	\vdots

ve které lze rychle vyhledávat podle prvního sloupce.

- ▶ Pro každý klíč $k'_2 \in \mathcal{K}_2$ spočteme $D_2(k'_2, y)$ a výsledek se pokusíme vyhledat v prvním sloupci tabulky.
- ▶ Pokud se výsledek podaří najít, pak máme

$$D_2(k'_2, y) = E_1(k'_1, x),$$

čili $y = E_2(k'_2, E_1(k'_1, x))$.

Meet-in-the-middle útok

- ▶ Časová složitost útoku je $|\mathcal{K}_1| + \frac{1}{2}|\mathcal{K}_2|$ operací.
 - ▶ Sestrojení tabulky: $|\mathcal{K}_1|$ volání operace E_1 .
 - ▶ Vyhledávání v tabulce: $\frac{1}{2}|\mathcal{K}_2|$ volání operace D_2 .
- ▶ Paměťová složitost útoku je minimálně $|\mathcal{K}_1| \log_2 |\mathcal{K}_1|$ bitů (pro uložení pravého sloupce tabulky).
- ▶ Složitost útoku hrubou silou na $S_2 \times S_1$ by byla $\frac{1}{2}|\mathcal{K}_1| \cdot |\mathcal{K}_2|$ operací.
- ▶ Meet-in-the-middle útok vyměňuje časovou složitost za paměťovou. Tuto vlastnost nazýváme *time-memory tradeoff*.

Double DES

- ▶ Double DES je šifra s klíčem o délce 112 bitů definovaná

$$2DES((k_1, k_2), x) := DES(k_2, DES(k_1, x)),$$

pro $k_1, k_2 \in \mathcal{K}_{DES}$ a $x \in \mathcal{P}_{DES}$.

- ▶ Jedná se o součin šifer a ví se, že DES není idempotentní.
- ▶ Meet-in-the-middle útok na 2DES:
 - ▶ Časová složitost útoku je $2^{56,6}$ šifrovacích operací.
 - ▶ Paměťová složitost útoku je minimálně 458 752 TB.
- ▶ Z hlediska časové složitosti útoků tedy není 2DES o nic bezpečnější než DES.
- ▶ Paměťovou složitost lze flexibilně snížit pomocí rainbow tables, ale zvýší se tím časová složitost.

Triple DES

- ▶ Triple DES definujeme jako

$$3DES((k_1, k_2, k_3), x) := DES(k_3, DES^{-1}(k_2, DES(k_1, x))),$$

pro $k_1, k_2, k_3 \in \mathcal{K}_{DES}$ a $x \in \mathcal{P}_{DES}$.

- ▶ Klíče k_1 , k_2 a k_3 lze volit třemi způsoby:

1. Three-key triple DES:

k_1 , k_2 a k_3 jsou nezávislé klíče (168 bitů).

2. Two-key triple DES:

k_1 a k_2 jsou nezávislé klíče a $k_3 = k_1$ (112 bitů).

3. Klasický DES:

$k_1 = k_2 = k_3$ (56 bitů).

- ▶ Proč $DES \times DES^{-1} \times DES$ spíše než $DES \times DES \times DES$?

- ▶ Zpětná kompatibilita. Máme-li hardwarovou implementaci 3DESu, můžeme pomocí ní počítat i DES ($k_1 = k_2 = k_3$).

Útoky na triple DES

- ▶ Meet-in-the-middle útok na two-key nebo three-key 3DES:
 - ▶ Triple DES můžeme vyjádřit jako součin dvou šifer: $(DES \times DES^{-1})$ a DES.
 - ▶ Časová složitost útoku je $2^{56} + 2^{111} \approx 2^{111}$ šifrovacích operací.
 - ▶ Paměťová složitost útoku je minimálně 458 752 TB.
- ▶ Oorschotův-Wienerův útok na two-key triple DES (1990):
 - ▶ Jedná se o known plaintext attack, ve kterém útočník zná 2^t párů OT a ŠT pro libovolné $t > 0$.
 - ▶ Časová složitost útoku je 2^{120-t} operací šifrování.
 - ▶ Paměťová složitost útoku je $O(2^t)$.
- ▶ Od roku 2016 NIST nedoporučuje používat two-key triple DES pro šifrování.

DES-X

- ▶ V roce 1984 navrhl Ron Rivest zesílení šifry DES přidáním tzv. bělicích klíčů:

$$\text{DES-X}((k_1, k_2, k_3), x) := k_3 \oplus \text{DES}(k_2, x \oplus k_1),$$

kde $x, k_1, k_3 \in \{0, 1\}^{64}$ a $k_2 \in \mathcal{K}_{\text{DES}}$.

- ▶ Klíč tak má celkem $64 + 56 + 64 = 184$ bitů.
- ▶ KPA hrubou silou má složitost jen 2^{120} operací šifrování. Stačí procházet k_1 a k_2 ($64 + 56$ bitů) a dopočítat k_3 .
- ▶ Bělení zlepšuje odolnost proti lineární i diferencíální kryptoanalýze na 2^{60} známých resp. volených párů OT a ŠT.
- ▶ Biryukovův-Wagnerův slide attack (2000):
 - ▶ Known plaintext útok, ve kterém útočník zná $2^{32,5}$ párů.
 - ▶ Časová složitost útoku je $2^{87,5}$ operací šifrování.