Blokové symetrické šifry, DES a jeho kryptoanalýza

Autor: Martin Kopecký, FJFI ČVUT
Úvod do kryptologie

13. 3. 2012

1. Blokové symetrické šifry
   - Symetrické vs. asymetrické šifry
   - Operační módy blokových šifer
   - Druhy útoků na kryptosystémy

2. DES
   - S-DES
   - DES a jeho kryptoanalýza
   - Další blokové šifry

Symetrické vs. asymetrické šifry

Vzpomeňme Caesarovu šifru, pár klíčů *(key-pair)* (e, d):

$$E_e(m) = m + 3(mod 26), \quad D_d(c) = c - 3(mod 26)$$

Symetrické (single-key, private key)

- „stejný“ klíč k šifrování i dešifrování
- klíč je tajný

\+ nízká výpočetní náročnost

– odesílatel a příjemce musí být předem domluveni na stejném klíči

Asymetrické (public key)

- privátní (tajný) a veřejný klíč *(public key)*
- veřejný klíč slouží k šifrování, privátní k dešifrování

– vysoká výpočetní náročnost

\+ odpadá nutnost výměny klíčů; když je třeba šifrovat, použije se veřejně dostupný klíč

Blokové vs. proudové šifry

Blokové šifry (block cipher)

- typ symetrické šifry
- zdrojový text rozdělí do bloků délky $k \epsilon N$, do posledního bloku se umístí tzv. výplň *(padding)*, $k = 64$ nebo 128 bitů
- každý blok je zakódován pomocí symetrického klíče
- nebezpečí použití stejného klíče na všechny bloky $=>$ různé druhy režimu blokové šifry *(modes of operation)*

Proudové šifry (stream cipher)

- další typ symetrické šifry
- otevřený text je kombinován s pseudonáhodným proudem bitů *(keystream)* — nejčastěji $XOR = \oplus$

\+ rychlejší než blokové šifry

− náchylnější k útokům, stejná šifra nesmí být použita dvakrát

Faktory při volbě metody šifrování

- rychlost šifrování a dešifrování
- schopnost samosynchronizace
- bezpečnost metody za předpokladu užití kvalitních klíčů

Šifrování HDD se systémem FAT aneb „Jak to nedělat“

- první dvě FAT tabulky mají stejný obsah
- první tabulka šifrovaná 512B klíčem k, druhá nešifrována, tj.

$$FAT_1 = FAT_2 \oplus k$$

- zbylé 512B sektory šifrovány stejným klíčem k
- druhá tabulka se nešifruje z kontrolních důvodů

Kde je chyba?

- klíč je možné triviálně získat:

$$k = FAT1 \oplus FAT2$$

- stejný klíč pro všechny zbývající bloky HDD

Elektronická kódová kniha (ECB)

- bloky otevřeného textu šifrujeme stejným klíčem nezávisle na sobě
- používá se pro krátké zprávy

– dva stejné bloky otevřeného textu => stejné bloky šifrového textu
útočník může bloky šifrového textu vyměňovat, vkládat nebo vyjímat:
$m =$ „... převeďte 1 000 Kč ...“ => „... převeďte 1 000 000 Kč ...“

$$\forall j \in \{1, 2, .., n\} \quad \begin{array}{l} c_j = E_k(m_j) \\ m_j = E_k^{-1}(c_j) \end{array}$$

Zřetězení šifrového textu (CBC)

- nejčastější, vhodný k použití s blokovými šiframi s veřejným klíčem
- přidává závislost právě šifrovaného bloku na předchozím
- vyžaduje padding
- *inicializační vektor IV* – nemusí být tajný

\+ stejné otevřené bloky mají odlišné šifrové obrazy

\+ samosynchronizující ve smyslu záměny bitů, či ztráty celých bloků

– pouze sekvenční zápis a čtení

– po přidání/smazání jednotlivých bitů se nezotaví

$$c_0 = IV$$

$$\forall j \in \{1, 2, .., n\} \quad \begin{array}{l} c_j = E_k(m_j \oplus c_{j-1}) \\ m_j = E_k^{-1}(c_j) \oplus c_{j-1} \end{array}$$

Zpětná vazba z šifrového textu (CFB)

- používá lineární posuvný registr, umožňuje zpracování po blocích velikosti $r < n$, kde n je délka bloku symetrické šifry
- ke zotavení je třeba n/r bloků
- 2 varianty: starší: $r < n$, novější: $r = n$
- nevhodný k použití s blokovými šiframi s veřejným klíčem
- převádí blokovou šifru na proudovou
- po přidání/smazání jednotlivých bitů se nezotaví

$$c_0 = IV$$

$$\forall j \in \{1, 2, .., n\} \quad k_j = E_k(c_{j-1}) \; \forall j \in \{1, 2, .., n\}$$
$$c_j = m_j \oplus k_j \; \forall j \in \{1, 2, .., n\}$$

Zpětná vazba z výstupu (OFB)

- podobné CFB, jen klíče jsou nezávislé na vstupním textu
- umožňuje paralelní zpracování - ze znalosti IV a k vypočte všechny klíče
- nevyžaduje padding
- převádí blokovou šifru na proudovou

\+ změna bitu v c_j ovlivní pouze bit na odpovídající pozici m_j

− po přidání/smazání jednotlivých bitů se nezotaví

$$k_0 = IV$$
$$\forall j \in \{1, 2, .., n\} \quad k_j = E_k(k_{j-1}) \; \forall j \in \{1, 2, .., n\}$$
$$c_j = m_j \oplus k_j \; \forall j \in \{1, 2, .., n\}$$

Pozor: nebezpečí použití stejného IV pro dva různé otevřené texty

$$c_i \oplus c_i' = m_i \oplus k_i \oplus m_i' \oplus k_i' = m_i \oplus m_i'$$

Čítačový mód (CTR)

- už v roce 1980, ale standardizovaný až 2001
- využíván u vysokorychlostních přenosů, např. protokol *IPSec*
- *IV* naplní blok čítače, který se zvyšuje o vhodnou konstantu; klíč vzniká šifrováním čítače
- umožňuje vypočítat klíč na libovolné pozici proudu
- odolnost vůči chybám stejná jako u OFB
- nevyžaduje padding

Pozor: musí se dodržet zásada, aby ani v jedné zprávě, ani v dalších zprávách šifrovaných stejným klíčem, nedošlo k vygenerování stejného obsahu čítače (tzv. *dvojí použití hesla*)

$$CTR_j = (IV + j) \ mod \ 2^B$$
$$k_j = E_k(CTR_j)$$
$$\forall j \in \{1, 2, .., n\}$$
$$c_j = m_j \oplus k_j$$
$$m_j = c_j \oplus k_j$$

Zabezpečovací kód zprávy (MAC)

- řeší otázku autentizace (obrana proti úmyslným nebo náhodným chybám)
- otevřený text je zašifrován libovolným módem
- další klíč mk, který slouží k ověření neporušenosti dat (něco jako kontrolní součet)
- MAC se vypočte CBC módem s parametry:

$$IV = 0, k = mk, \ \forall j \in \{1, 2, .., n\} \ c_j = E_k(m_j \oplus c_{j-1})$$
$$MAC := c_n$$

- z výsledného MAC se obvykle bere jen několik málo posledních bitů (taková část, která vytvoří odolný zabezpečovací kód)

– kombinace CBC a MAC je systémově náročná, zejména na paměť => potřeba nových operačních módů, které by řešily utajení i autentizaci najednou

Druhy útoků na kryptosystémy

„Def.“ Útok na kryptosystém

V podstatě jakákoliv metoda, u které máme nějakou počáteční informaci o otevřeném textu a patřičný šifrový text, který je zašifrován neznámým klíčem.

Pasivní

- kryptoanalytik pouze sleduje přenosový kanál
- ohrožuje pouze důvěrnost dat

Aktivní

- kryptoanalytik upravuje posílané zprávy
- ohrožuje důvěrnost dat, jejich integritu a pravost (autenticitu)

Pasivní útoky

1. Útok s možností volby otevřeného textu *(Chosen-plaintext)*
   - zvolíme otevřený text, dáme ho nepříteli zašifrovat a získáme šifrový text; porovnáním textů získáme klíč
2. Útok s možností volby šifrového textu *(Chosen-ciphertext)*
   - zvolíme šifrový text, dáme ho nepříteli dešifrovat a získáme otevřený text
3. Útok se znalostí otevřeného textu *(Known-plaintext)*
   - máme k dispozici několik párů otevřený text — šifrový text
4. Útok se znalostí šifrového textu *(Cipher-text only)*
   - máme k dispozici pouze šifrový text
5. Adaptivní útok s možností volby otevřeného/šifrového textu
   - 1./2. metoda s tím, že volba vstupního textu je ovlivněna předchozím výstupním

Aktivní útoky

1. Útok hrubou silou *(Brute-Force Attack)*
   - úplné prohledávání prostoru klíčů
2. Slovníkový útok *(Dictionary Attack)*
   - útok hrubou silou s posloupností pravděpodobných hesel
   - rychlejší ale méně efektivní
3. Narozeninový útok *(The Birthday Attack)*
   - název pochází z teorie pravděpodobnosti — *narozeninový problém*
   - snaží se nalézt dvě vstupní hodnoty x_1 a x_2 pro hašovací fci f takové, že $f(x_1) = f(x_2)$, tzv. *kolize*
   - Alice bude mít dvě zprávy — m a m', které budou mít stejný otisk $f(m) = f(m')$. Bob podepíše zprávu m, ale Alice ji potom vymění za m'. Potom se bude zdát, že Bob podepsal zprávu m'.

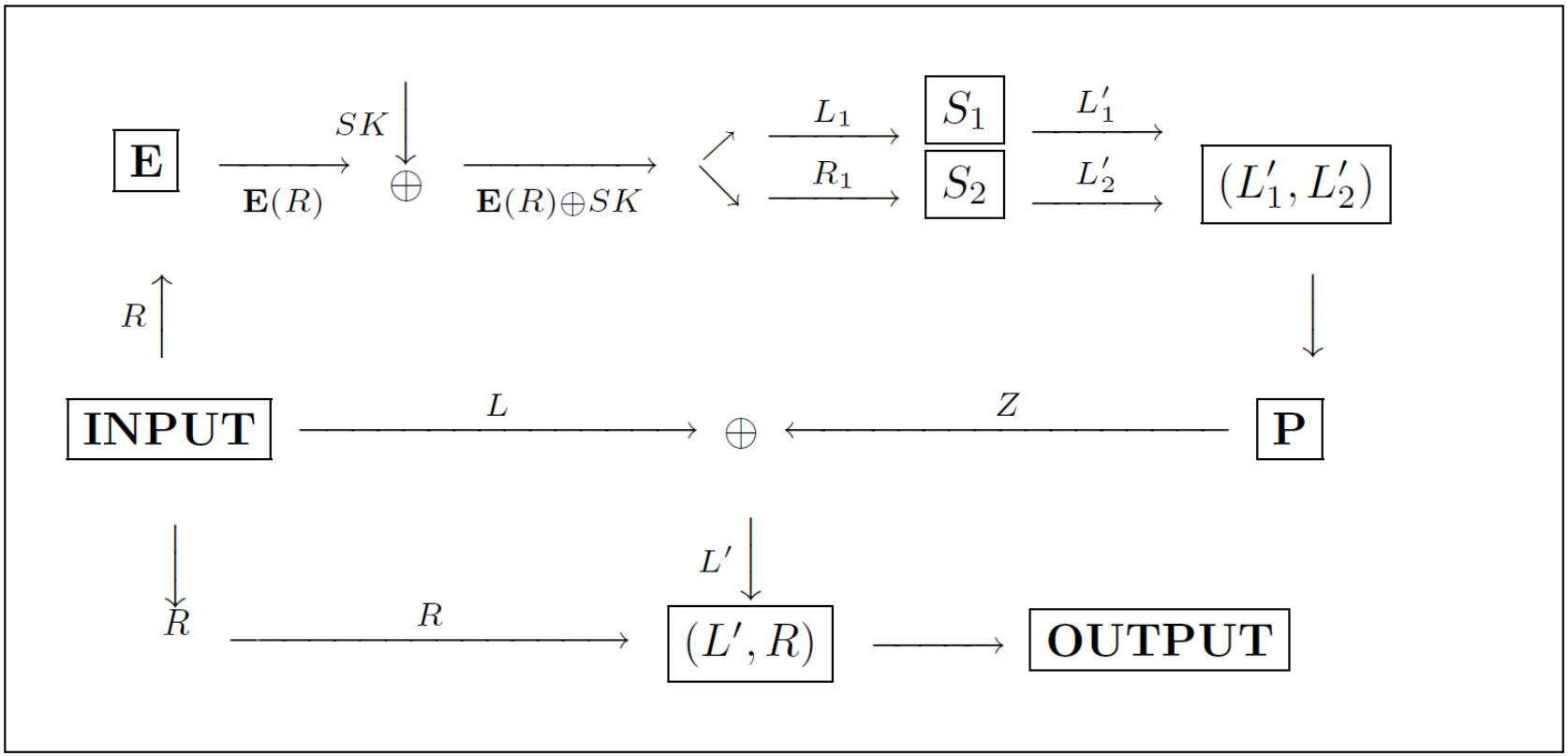
Aktivní útoky

4. Útok postranním kanálem *(Side Channel Attack)*
   - zneužívá informace, které unikají přímo z fyzické implementace systému během běhu kryptografického algoritmu
   - zjistí klíč nebo používaný algoritmus na základě časové náročnosti výpočtu, měření spotřeby zařízení při výpočtu (zvláště u čipových karet), EM záření zářízení nebo zaváděním HW chyb (změna napětí, extrémní teplota, ozáření tranzistorů $=>$ provede se jiná větev podmínky IF)

S-DES (Simplified DES)

- zjednodušená verze DES pro názornost
- délka m je 8 bitů, délka k 10 bitů, 2 kolečka *(rounds)*

S-DES round



S-DES algoritmus

1. aplikujeme inicializační permutaci na otevřený text $IP(m) = INPUT$
2. 8mi bitový $INPUT$ rozdělíme na 4 levé L a 4 pravé bity R
3. rozšíříme R expanzní permutací E
4. sečteme $E(R)$ modulo 2 s 8mi bitovým subklíčem SK
5. výsledek rozdělíme na dva 4bitové bloky (L_1, R_1)
6. na L_1 i R_1 aplikujeme S-BOXy a výsledky (2bitové) slepíme do 4bitového (L'_1, L'_2)
7. aplikujeme permutaci $P(L'_1, L'_2) = Z$
8. výsledek Z přičteme modulo 2 k $L => L'$
9. výsledek kolečka je potom 8bitový $(L', R) = OUTPUT$
10. u získaného $OUTPUTu$ prohodíme levou a pravou stranu $= INPUT$ pro druhé kolečko — kroky 3–10
11. nově získané (L', R) permutujeme $IP^{-1} =>$ šifrový text c

S-BOX (Substitution BOX)

- 4bitový vstup, 2bitový výstup
- „nabourává“ linearitu DES
  - Def. *lineární šifra*: každý výstupní bit je lineární kombinací vstupních bitů

S_1		x_2	0	0	1	1
		x_3	0	1	0	1
x_1	x_4					
0	0		01	00	11	10
0	1		11	10	01	00
1	0		00	10	01	11
1	1		00	01	11	10

S_2		x_2	0	0	1	1
		x_3	0	1	0	1
x_1	x_4					
0	0		00	01	10	11
0	1		10	00	01	11
1	0		11	00	01	10
1	1		10	01	00	11

Příklad: $S_1(1101) = 11$, $S_2(1101) = 00$

DES (Data Encryption Standard)

- 1977 USA — šifrovací standard pro ochranu citlivých neutajovaných dat ve státní správě
- délka m je 64 bitů, délka k 56 bitů (generuje se z nich 16 48mi bitových podklíčů *(subkeys)*), 16 koleček; už při vzniku bylo upozorňováno na příliš krátký klíč, ale IBM byla nucena vlivem NSA zanést do návrhu 56 bitů
- někdy se uvádí 64 bitů, ale: nejnižší bit v bajtu = lichá parita horních 7 bitů
- slabé klíče — $(\exists 4 k \epsilon K)(E_k(m) = m)$
- poloslabé klíče — $(\exists 6 (k_1, k_2) \epsilon (K, K))(E_{k_2}(E_{k_1}(m)) = m)$
- komplementárnost — $(\forall k \epsilon K)(\forall m \epsilon M)(E_k(m) = !(E_{!k}(!m)))$
- lineární a diferenciální kryptoanalýza

Kryptoanalýza DES

- DES hodně diskutován
- 1977 Diffie a Hellman — teoretický útok hrubou silou na stroji s 10^6 čipy, \$20 mil., doba luštění v průměru 12 hodin
- 1993 Wiener — návrh *DES cracker* = stroj s 57600 DES čipy, \$1 mil., doba luštění v průměru 3,5 hodiny

DES Challenges

- veřejné soutěže s úkolem nalézt klíč metodou hrubé síly, odměna \$10000
- cíl: veřejně předvést nízkou odolnost DES (do té doby pouze předpokládána odbornou veřejností)

DES Challenges

DES-I (1997)

- projekt *DESCHALL* (Rocke Verser, Matt Curtin a Justin Dolske) — distribuované výpočty přes internet
  - 96 dní po zahájení hledání se klíč našel, prohledáno 25% prostoru

$\Rightarrow$ možnost provádění distribuovaných výpočtů výrazně zvyšuje požadavky na odolnost algoritmů

DES-II-1 (1998)

- distributed.net — řešení za 41 dní, prohledáno 85% prostoru

DES-II-2 (1998)

- stroj *Deep Crack* od spol. EFF, $250000 — doba luštění v průměru 4,5 dne, řešení se našlo za 56 hodin

DES-III (1999)

- distributed.net a EFF dohromady — řešení za 22 hodin

Deep Cracker a Paul Kocher

3DES (TripleDes)

= 3 x DES po sobě — se třema nebo dvěma klíči:
$c = E_{k_3}(D_{k_2}(E_{k_1}(m)))$ nebo $c = E_{k_1}(D_{k_2}(E_{k_1}(m)))$

- délka klíče 112 *(Double DES)* nebo 168 bitů

\+ bezpečná

\− pomalá

Další blokové šifry

Protože šifrování pomocí 3DES bylo pomalé, začaly se používat další blokové šifry. Všechny využívaly toho, že stroje byly stavěné na 64 bitovou délku klíče (vzpomeňte 8 paritních bitů z 64 u DES)

- CAST
- IDEA
- RC2
- Blowfish
- Twofish
- ...

AES *(Advanced Encryption Standard)*

- 2. 1. 1997 NIST (americký standardizační úřad) vypsal výběrové řízení na novou šifru pojmenovanou AES
- přihlásilo se 15 kandidátů, ve finále 5
- 26. 11. 2001 vyhrál Rijndael (autoři Joad Daemen a Vincent Rijmen)
- 26. 5. 2002 federální standard USA
- délka klíče 128, 192, 256 bitů
- počet koleček 10, 12, 14
- velikost bloku 128 bitů
- „128 bitů zaručuje utajení minimálně na několik desítek let dopředu“

Zdroje

- Richard A. Mollin. An Introduction to Cryptography (second edition). Chapman & Hall/CRC, 2007.
- Vlastimil Klíma. Seriál Kryptologie pro praxi, Sdělovací technika 2003–2011.
- http://www.wikipedia.org
- http://www.karlin.mff.cuni.cz/~tuma/ciphers08/sifry5.ppt
- http://kmlinux.fjfi.cvut.cz/~balkolub/Vyuka/leto2011/Havlicek.pdf
- http://www.pocitacovysvet.ic.cz/Pocitacovy%20utok/kryptograficky.utok.menu.html
- http://www.kohout.se/files/bezpecnost_des.pdf

Děkuji za pozornost.