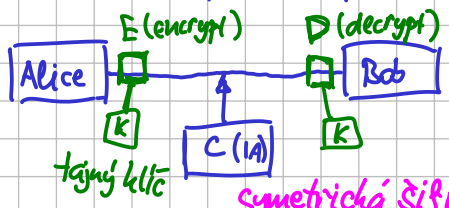
Šifrovaný přenos zpráv:



symetrická šifra

Praktické příklady: AES (128–256b klíč)
ChaCha20 (256b)

Předpoklady:

- E, D jsou veřejně známé
- klíč je tajný
  - ↳ dnešní typicky: klíče má 256 bitů
    ⇓
    # klíčů = $2^{256} = 10^{3 \cdot 25{,}6} \doteq 10^{78}$
    $2^{10} = 1024 \doteq 10^3$
- délka zprávy není tajná
- zprávy mají být smysluplné

Caesarova šifra

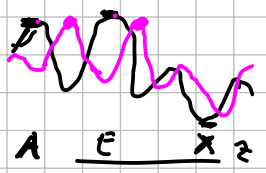
0	1	2	…	25
A	B	C	D E …	W X Y Z
D	E	F	G H	Z A B C

$E_k(x) = (x + 3) \bmod 26$ (k = 3)

$D_k(x) = (x - 3) \bmod 26$

problémy:
- málo klíčů (26)
- pozice ve zprávě se nezměňuje

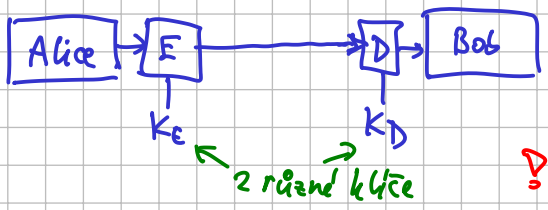
histogram četností



statistická analýza

10^8 klíčů = 100M
CPU: ~10^9 operací/s
10^{12} klíčů ~ 1000 sekund

asymetrická šifra



generuji (K_E, K_D)

? pomalé...

Šifrovaný e-mail:
- každý si vygeneruje svůj pár klíčů (K_E, K_D) a K_E zveřejní (K_D zůstává soukromé)

Podpisy: naopak K_D je veřejný a K_E soukromý

Trik: Hybrid sym. a asym. šifry

Alice: zpráva x, náhodný klíč K_{sym}
$y = E_{sym, K_{sym}}(x)$
$z = E_{asym, K_{E\_Bob}}(K_{sym})$
(y, z) → Bob

Příklad: RSA

zprávy jsou čísla počítáme mod m (součin dvou prvočísel)

exponenty e, d

$K_E = (e, m)$ $K_D = (d, m)$

$E(x) = x^e \bmod m$

$D(x) = x^d \bmod m$

$2^{100} \bmod m = ?$

Více podpisů 1 zprávy:

① x → E_{K_1} → E_{K_2} → … → E_{K_n} → podepsaná x

② x → E_{K_1}, …, E_{K_n} → klepit

trik: podepisuji H(zpráva)

kryptografická hešovací funkce

zpráva → H → otisk pevné velikosti (např. 256b)

příklad: SHA-256

chci: nejde inverzovat, nedokážu najít kolizi: $H(x) = H(x')$

autentifikace: A = uživatel, B = server
A chce přesvědčit B, že zná heslo

A ← H(heslo ‖ n) — n náhodný řetězec bitů — B

ukládání hesel na serverech:
pro uživatele: náhodná sůl

n, sůl, $H(heslo \| sůl)$

A ← $H(H(heslo \| sůl)_B \| n)$

Vernamova šifra (one-time pad)

zpráva x_1 —— x_n bity
klíč k_1 —— k_n náhodný na 1 použití

výstup y_1 —— y_n

$y_i = x_i \text{ XOR } k_i$

dešifrování: $y_i \text{ XOR } k_i = x_i$

Opakování klíče: $y = x \text{ XOR } k$
$y' = x' \text{ XOR } k$

$y \text{ XOR } y' = (x \text{ XOR } k) \text{ XOR } (x' \text{ XOR } k) =$
$= x \text{ XOR } x'$

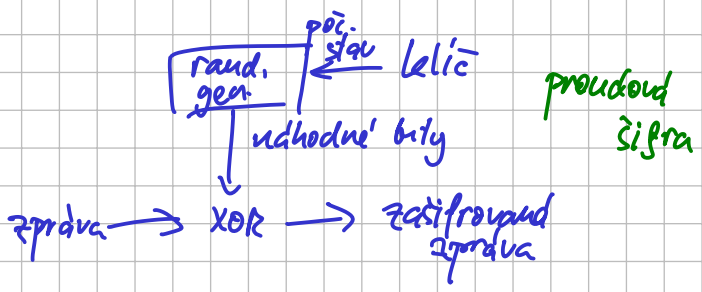
Kombinace s pseudonáhodným generátorem:

XOR	0	1
0	0	1
1	1	0

Tvrzení: Pokud $k_1 - k_n$ je náhodné, pak $y_1 - y_n$ je náhodné.

Důkaz: Pokud $x_i = 0$ (1), pak
- Pr 1/2: $k_i = 0 \to y_i = 0$ (1)
- Pr 1/2: $k_i = 1 \to y_i = 1$ (0)

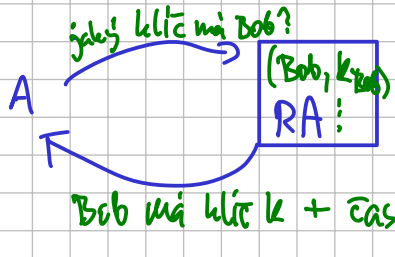
Ať je x_i cokoliv, y_i je
- 0 Pr = 1/2
- 1 Pr = 1/2



Problém distribuce klíčů

Jak Alice zjistí Bobův veřejný klíč? (bezpečně)

① registrační autorita, které všichni věří a znají její veřejný klíč



② certifikační autorita
- všichni znají její veřejný klíč
- generuje certifikáty: (Bob, H(klíč), platí od do, podpis)

Bob při navázání spojení pošle:
Bobův veř. klíč
certifikát