Vigenèrejeva šifra (1586):

Naj bo $m \in \mathbb{N}$ in

$$\mathcal{P} = \mathcal{C} = \mathcal{K} = (\mathbb{Z}_{26})^m.$$

Za ključ $K = (k_1, k_2, \ldots, k_m)$ definiramo

$$\begin{aligned} e(x_1, \ldots, x_m) &= (x_1 + k_1, \ldots, x_m + k_m) \text{ in} \\ d(y_1, \ldots, y_m) &= (y_1 - k_1, \ldots, y_m - k_m), \end{aligned}$$

kjer sta operaciji "+" in "−" opravljeni po modulu 26.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
A	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
B	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A
C	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B
D	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C
E	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D
F	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E
G	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F
H	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G
I	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H
J	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I
K	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J
L	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K
M	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L
N	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M
O	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N
P	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
Q	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
R	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
S	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
T	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
U	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
V	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
W	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V
X	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
Y	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X
Z	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y

Sporočilo

```
TO BE OR NOT TO BE THAT IS THE QUESTION
```

zašifriramo s ključem `RELATIONS`:

```
ključ:       RELAT IONSR ELATI ONSRE LATIO NSREL
čistopis:    TOBEO RNOTT OBETH ATIST HEQUE STION
tajnopis:    KSMEH ZBBLK SMEMP OGAJX SEJCS FLZSY
```

Npr. prvo črko tajnopisa dobimo tako, da pogledamo v tabelo na mesto (`R`, `T`).

Kako pa najdemo iz `T` in `K` nazaj `R`?

To ni monoabecedna šifra.

Pravimo ji **poliabecedna šifra**.

Vigenèrejeva šifra in 26^m možnih ključev.

Za $m = 5$ je število 1.1×10^7 že preveliko, da bi "peš" iskali pravi ključ.

Hillova šifra (1929)

Naj bo m neko naravno število in naj bo

$$\mathcal{P} = \mathcal{C} = (\mathbb{Z}_{26})^m.$$

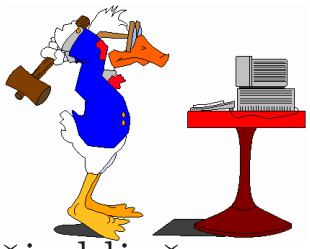
Za K vzemimo obrnljivo $m \times m$ matriko in definirajmo

$$e_K(x) = xK \quad \text{in} \quad d_K(y) = yK^{-1},$$

pri čemer so vse operacije opravljene v $\mathbb{Z}_{26}$.

Ponovimo:

Odšifriranje (razbijanje) klasičnih šifer



Kriptografske sisteme kontroliramo s pomočjo ključev, ki določijo transformacijo podatkov.
Seveda imajo tudi ključi digitalno obliko (binarno zaporedje: 010011010101...).

Držali se bomo **Kerckhoffovega principa**, ki pravi, da "nasprotnik"

pozna kriptosistem oziroma algoritme,
ki jih uporabljamo, ne pa tudi ključe,
ki nam zagotavljajo varnost.

Ločimo naslednje nivoje napadov na kriptosisteme:

1. **samo tajnopis**: *nasprotnik ima del tajnopisa,*
2. **poznani čistopis**: *nasprotnik ima del čistopisa ter ustrezen tajnopis,*
3. **izbrani čistopis**: *nasprotnik ima začasno na voljo šifrirno mašinerijo ter za izbrani $x \in \mathcal{P}$ konstruira $e(x)$,*
4. **izbrani tajnopis**: *nasprotnik ima začasno na voljo odšifrirno mašinerijo ter za izbrani $y \in \mathcal{C}$ konstruira $d(y)$.*

Odšifriranje Vigenèrejeve šifre

Test Friedericha Kasiskega (1863):

(in Charles Babbage-a 1854)

poiščemo dele tajnopisa $\boldsymbol{y} = y_1 y_2 \ldots y_n$, ki so identični in zabeležimo razdalje $d_1, d_2, \ldots$ med njihovimi začetki. Predpostavimo, da iskani m deli največji skupni delitelj teh števil.

Naj bo $d = n/m$. Elemente tajnopisa $\boldsymbol{y}$ zapišemo po stolpcih v $(m \times d)$-razsežno matriko. Vrstice označimo z $\boldsymbol{y_i}$, tj.

$$\boldsymbol{y_i} \;=\; y_i \, y_{m+i} \, y_{2m+i} \, \ldots$$

Indeks naključja (William Friedman, 1920):

Za zaporedje $\boldsymbol{x} = x_1 x_2 \ldots x_d$ je **indeks naključja** (angl. index of coincidence, oznaka $I_c(\boldsymbol{x})$) **verjetnost**, da sta naključno izbrana elementa zaporedja $\boldsymbol{x}$ enaka.

Če so $f_0, f_1, \ldots, f_{25}$ frekvence črk $A, B, \ldots, Z$ v zaporedju $\boldsymbol{x}$, je

$$I_c(\boldsymbol{x}) \quad = \quad \frac{\sum_{i=0}^{25} \binom{f_i}{2}}{\binom{d}{2}} \quad = \quad \sum_{i=0}^{25} \frac{f_i(f_i-1)}{d(d-1)}.$$

Če so p_i pričakovane verjetnosti angleških črk, potem je

$$I_c(\boldsymbol{x}) \approx \sum_{i=0}^{25} p_i^2 = 0.065.$$

Za povsem naključno zaporedje velja

$$I_c(\boldsymbol{x}) \approx 26\left(\frac{1}{26}\right)^2 = \frac{1}{26} = 0.038.$$

Ker sta števili .065 in .038 dovolj narazen, lahko s to metodo najdemo dolžino ključa

(ali pa potrdimo dolžino, ki smo jo uganili s testom Kasiskega).

Za podzaporedje $\boldsymbol{y_i}$ in $0 \leq g \leq 25$ naj bo

$$M_g(\boldsymbol{y_i}) \;=\; \sum_{i=0}^{25} p_i \, \frac{f_{i+g}}{d}.$$

Če je $g = k_i$, potem pričakujemo

$$M_g(\boldsymbol{y_i}) \;\approx\; \sum_{i=0}^{25} p_i^2 \;=\; 0.065$$

Za $g \neq k_i$ je običajno M_g bistveno manjši od 0.065.

Torej za vsak $1 \leq i \leq m$ in $0 \leq g \leq 25$ tabeliramo vrednosti M_g, nato pa v tabeli za vsak $1 \leq i \leq m$ poiščemo tiste vrednosti, ki so blizu 0.065.

Ustrezni g-ji nam dajo iskane zamike $k_1, k_2, \ldots, k_m$.

Odšifriranje Hillove šifre

Predpostavimo, da je nasprotnik določil m, ki ga uporabljamo, ter se dokopal do m različnih parov m-teric (2. stopnja – poznan čistopis):

$$x_j = (x_{1,j}, x_{2,j}, ..., x_{m,j}),\ y_j = (y_{1,j}, y_{2,j}, ..., y_{m,j}),$$

tako da je $y_j = e_K(x_j)$ za $1 \leq j \leq m$.

Za matriki $X = (x_{i,j})$ in $Y = (y_{i,j})$ dobimo matrično enačbo $Y = XK$.

Če je matrika X obrnljiva, je $K = YX^{-1}$.

Za Hillovo šifro lahko uporabimo tudi 1. stopnjo napada (samo tajnopis), glej nalogo 1.25.

Koliko ključev imamo na voljo v primeru Hillove šifre? Glej nalogo 1.12.

Za afino-Hillovo šifro glej nalogo 1.24.

Tokovne šifre

Naj bo $x_1x_2\ldots$ čistopis.

Doslej smo obravnavali kriptosisteme z enim samim ključem in tajnopis je imel naslednjo obliko.

$$\mathbf{y} = y_1y_2\cdots = e_K(x_1)e_K(x_2)\ldots$$

Taki šifri pravimo **bločna šifra**
(angl. block cipher).

Posplošitev: iz enega ključa $K \in \mathcal{K}$ napravimo zaporedje (tok) ključev. Naj bo f_i funkcija, ki generira i-ti ključ:

$$z_i = f_i(K, x_1, \ldots, x_{i-1}).$$

Z njim izračunamo:

$$y_i = e_{z_i}(x_i) \qquad \text{in} \qquad x_i = d_{z_i}(y_i).$$

Bločna šifra je poseben primer tokovne šifre (kjer je $z_i = K$ za vse $i \geq 1$).

Sinhrona tokovna šifra *je sedmerica* $(\mathcal{P}, \mathcal{C}, \mathcal{K}, \mathcal{L}, \mathcal{F}, \mathcal{E}, \mathcal{D})$ *za katero velja:*

1. $\mathcal{P}$ *je končna množica možnih* **čistopisov**,
2. $\mathcal{C}$ *je končna množica možnih* **tajnopisov**,
3. $\mathcal{K}$ *je končna množica možnih* **ključev**,
4. $\mathcal{L}$ *je končna množica tokovne* **abecede**,
5. $\mathcal{F} = (f_1, f_2, \dots)$ *je generator toka ključev:*

$$f_i : \mathcal{K} \times \mathcal{P}^{i-1} \longrightarrow \mathcal{L} \quad \text{za } i \geq 1$$

6. *Za vsak ključ* $z \in \mathcal{L}$ *imamo šifrirni* $(e_z \in \mathcal{E})$ *in odšifrirni* $(d_z \in \mathcal{D})$ *postopek, tako da je* $d_z(e_z(x)) = x$ *za vsak* $x \in \mathcal{P}$.

Za šifriranje čistopisa $x_1x_2\ldots$ zaporedno računamo

$$z_1, y_1, z_2, y_2, \ldots,$$

za odšifriranje tajnopisa $y_1y_2\ldots$ pa zaporedno računamo

$$z_1, x_1, z_2, x_2, \ldots.$$

Tokovna šifra je **periodična** s periodo d kadar, je $z_{i+d} = z_i$ za vsak $i \geq 1$

(poseben primer: Vigenèrejeva šifra).

Začnimo s ključi $(k_1, \ldots, k_m)$ in naj bo $z_i = k_i$ za $i = 1, \ldots, m$.
Definiramo linearno rekurzijo stopnje m:

$$z_{i+m} = z_i + \sum_{j=1}^{m-1} c_j \, z_{i+j} \mod 2,$$

kjer so $c_1, \ldots, c_{m-1} \in \mathbb{Z}_2$ vnaprej določene konstante.

Za ustrezno izbiro konstant $c_1, \ldots, c_{m-1} \in \mathbb{Z}_2$ in neničelen vektor $(k_1, \ldots, k_m)$ lahko dobimo tokovno šifro s periodo $2^m - 1$.

Hitro lahko generiramo tok ključev z uporabo **LFSR** (**Linear Feedback Shift Register**).

V pomičnem registru začnemo z vektorjem

$$(k_1, \ldots, k_m).$$

Nato na vsakem koraku naredimo naslednje:

1. k_1 dodamo toku ključev (za XOR),
2. $k_2, \ldots, k_m$ pomaknemo za eno v levo,
3. 'nov' ključ k_m izračunamo z

$$\sum_{j=0}^{m-1} c_j \, k_{j+1} \quad \text{(to je "linear feedback")}.$$

Primer:

$c_0 = 1$, $c_1 = 1$, $c_2 = 0$, $c_3 = 0$,

torej je $k_{i+4} = k_i + k_{i+1}$.

Izberimo $k_0 = 1$, $k_1 = 0$, $k_2 = 1$, $k_3 = 0$.

Potem je $k_4 = 1$, $k_5 = 1$, $k_6 = 0$, ...

Naj bo $\boldsymbol{k} = (k_0, k_1, k_2, k_3)^t$ in

$$A := \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 \end{pmatrix}.$$

Torej je $A(\boldsymbol{k}) = (k_1, k_2, k_3, k_4)^t$,

$A^2(\boldsymbol{k}) = A(k_1, k_2, k_3, k_4)^t = (k_2, k_3, k_4, k_5)^t$

...

$A^i(\boldsymbol{k}) = (k_i, k_{i+1}, k_{i+2}, k_{i+3})^t$.

Najdaljša možna perioda je 15.

Enkrat dobimo:

$$A^i(\boldsymbol{k}) = A^j(\boldsymbol{k})$$

in ker je A obrnljiva

$$A^{i-j}(\boldsymbol{k}) = \boldsymbol{k}$$

Karakteristični polinom matrike A je

$$f(x) = 1 + x + x^4.$$

Ker je $f(x)$ nerazcepen, je $f(x)$ tudi minimalni polinom matrike A.

Red matrike A je najmanjše naravno število s, tako da je $A^s = I$. Naj bo e najmanjše naravno število, tako da $f(x) \mid (x^e - 1)$. Potem je $e = s$.

$$1 + x^{15} = (x+1)(x^2+x+1)(x^4+x+1)$$
$$(x^4+x^3+1)(x^4+x^3+x^2+x+1).$$

Splošno: če hočemo, da nam rekurzija stopnje m da periodo $2^m - 1$, potem si izberemo nerazcepen f.

Analiza je neodvisna od začetnega neničelnega vektorja.

Kriptoanaliza LFSR tokovne šifre:
uporabimo lahko poznan čistopis, glej nalogo 1.27.

2. poglavje

Shannonova teorija

- Popolna varnost
- Entropija
- Lastnosti entropije
- Ponarejeni ključi
  in enotska razdalja
- Produktne šifre

Popolna varnost

Omenimo nekaj osnovnih principov za študij varnosti nekega kriptosistema:

- **računska varnost**,
- **brezpogojna varnost**,
- **dokazljiva varnost**.

Kriptosistem je **računsko varen**, če tudi najboljši algoritem za njegovo razbitje potrebuje vsaj N operacij, kjer je N neko konkretno in zelo veliko število.

Napadalec (Oskar) ima na razpolago 18 Crayev, 4000 Pentium PC-jev in 200 DEC Alpha mašin (Oskar je "računsko omejen").

Kriptosistem je **dokazljivo varen** (angl. provable secure), če lahko pokažemo, da se njegova varnost zreducira na varnost kriptosistema, ki je zasnovan na dobro preštudiranem problemu.

Ne gre torej za absolutno varnost temveč *relativno varnost*.

Gre za podobno strategijo kot pri dokazovanju, da je določen problem *NP-poln* (v tem primeru dokažemo, da je dani problem vsaj tako težak kot nekdrugi znani NP-poln problem, ne pokažemo pa, da je absolutno računsko zahteven).

Kriptosistem je **brezpogojno varen**, kadar ga napadalec ne more razbiti, tudi če ima na voljo neomejeno računsko moč.

Seveda je potrebno povedati tudi, kakšne vrste napad imamo v mislih. Spomnimo se, da zamične, substitucijske in Vigenère šifre niso varne pred napadom s poznanim tajnopisom (če imamo na voljo dovolj tajnopisa).

Razvili bomo teorijo kriptosistemov, ki so brezpogojno varni pri napadu s poznanim tajnopisom. Izkaže se, da so vse tri šifre brezpogojno varne, kadar zašifriramo le en sam element čistopisa.

Glede na to, da imamo pri brezpogojni varnosti na voljo neomejeno računsko moč, je ne moremo študirati s pomočjo teorije kompleksnosti, temveč s teorijo verjetnosti.

Naj bosta X in Y slučajni spremenljivki,
naj bo $p(x) := P(X = x)$, $p(y) := P(Y = y)$ in
$p(x \cap y) := P((X = x) \cap (Y = y))$ produkt dogodkov.

Slučajni spremenljivki X in Y sta **neodvisni**, če in samo, če je $p(x \cap y) = p(x)p(y)$ za vsak $x \in X$ in $y \in Y$.

Omenimo še zvezo med pogojno verjetnostjo in pa verjetnostjo produkta dveh dogodkov oziroma **Bayesov izrek o pogojni verjetnosti**:

$$p(x \cap y) = p(x/y)p(y) = p(y/x)p(x),$$

iz katerega sledi, da sta slučajni spremenljivki X in Y neodvisni, če in samo, če je $p(x/y) = p(x)$ za vsak x in y.

Privzemimo, da vsak ključ uporabimo za največ eno šifriranje, da si Anita in Bojan izbereta ključ K z neko fiksno verjetnostno porazdelitvijo $p_{\mathcal{K}}(K)$ (pogosto enakomerno porazdelitvijo, ni pa ta nujna) in naj bo $p_{\mathcal{P}}(x)$ verjetnost čistopisa x.

Končno, predpostavimo, da sta izbira čistopisa in ključa neodvisna dogodka.

Porazdelitvi $\mathcal{P}$ in $\mathcal{K}$ inducirata verjetnostno porazdelitev na $\mathcal{C}$. Za množico vseh tajnopisov za ključ K

$$C(K) = \{e_K(x) \,|\, x \in \mathcal{P}\}$$

velja

$$p_{\mathcal{C}}(y) = \sum_{\{K \,|\, y \in C(K)\}} p_{\mathcal{K}}(K)\, p_{\mathcal{P}}(d_k(y))$$

in

$$P(Y = y / X = x) = \sum_{\{K \,|\, x = d_K(y)\}} p_{\mathcal{K}}(K).$$

Sedaj lahko izračunamo pogojno verjetnost $p_{\mathcal{P}}(x/y)$, tj. verjetnost, da je x čistopis, če je y tajnopis

$$P(X = x/Y = y) = \frac{p_{\mathcal{P}}(x) \times \sum\limits_{\{K \,|\, x = d_K(y)\}} p_{\mathcal{K}}(K)}{\sum\limits_{\{K \,|\, y \in C(K)\}} p_{\mathcal{K}}(K)\, p_{\mathcal{P}}(d_k(y))}$$

in opozorimo, da jo lahko izračuna vsakdo, ki pozna verjetnostni porazdelitvi $\mathcal{P}$ in $\mathcal{K}$.

Primer: $\mathcal{P} = \{a, b\}$ in $\mathcal{K} = \{K_1, K_2, K_3\}$:

$$p_{\mathcal{P}}(a) = 1/4 \text{ in } p_{\mathcal{P}}(b) = 3/4.$$

$$p_{\mathcal{K}}(K_1) = 1/2 \text{ in } p_{\mathcal{K}}(K_2) = p_{\mathcal{K}}(K_3) = 1/4.$$

Enkripcija pa je definirana z $e_{K_1}(a) = 1$, $e_{K_1}(b) = 2$; $e_{K_2}(a) = 2$, $e_{K_2}(b) = 3$; $e_{K_3}(a) = 3$, $e_{K_3}(b) = 4$.

Potem velja

$$p_{\mathcal{C}}(1) = \frac{1}{8}, \quad p_{\mathcal{C}}(2) = \frac{7}{16}, \quad p_{\mathcal{C}}(3) = \frac{1}{4}, \quad p_{\mathcal{C}}(3) = \frac{3}{16}.$$

$$p_{\mathcal{P}}(a/1) = 1, \; p_{\mathcal{P}}(a/2) = \frac{1}{7}, \; p_{\mathcal{P}}(a/3) = \frac{1}{4}, \; p_{\mathcal{P}}(a/4) = 0.$$

Šifra $(\mathcal{P}, \mathcal{K}, \mathcal{C})$ je **popolnoma varna**, če je

$$P(X = x/Y = y) = p_{\mathcal{P}}(x) \quad \text{za vse} \quad x \in \mathcal{P} \quad \text{in} \quad y \in \mathcal{C},$$

tj. "končna" verjetnost, da smo začeli s tajnopisom x pri danem čistopisu y, je identična z "začetno" verjetnostjo čistopisa x.

V prejšnjem primeru je ta pogoj zadoščen samo v primeru $y = 3$, ne pa tudi v preostalih treh.

Izrek 1. *Če ima vseh 26 ključev pri zamični šifri enako verjetnost* $1/26$*, potem je za vsako verjetnostno porazdelitev čistopisa zamična šifra popolnoma varna.*

Dokaz: $\mathcal{P} = \mathcal{C} = \mathcal{K} = \mathbb{Z}_{26}$, $e_K(x) = x + K \bmod 26$:

$$p_{\mathcal{C}}(y) = \frac{1}{26} \sum_{K \in \mathbb{Z}_{26}} p_{\mathcal{P}}(y - K) = \frac{1}{26},$$

$$P(Y = y / X = x) = p_{\mathcal{K}}(y - x \bmod 26)) = \frac{1}{26}. \quad \blacksquare$$

Torej lahko zaključimo, da zamične šifre ne moremo razbiti, če za vsak znak čistopisa uporabimo nov, naključno izbran ključ.

Sedaj pa preučimo popolno varnost na splošno. Pogoj

$$P(X = x/Y = y) = p_{\mathcal{P}}(x) \quad \text{za vse} \quad x \in \mathcal{P} \quad \text{in} \quad y \in \mathcal{C}$$

je ekvivalenten pogoju

$$P(Y = y/X = x) = p_{\mathcal{C}}(y) \quad \text{za vse} \quad x \in \mathcal{P} \quad \text{in} \quad y \in \mathcal{C}.$$

Privzemimo (BŠS), da je $p_{\mathcal{C}}(y) > 0$ za vse $y \in \mathcal{C}$. Ker je $P(Y = y/X = x) = p_{\mathcal{C}}(y) > 0$ za fiksen $x \in \mathcal{P}$ in za vsak $y \in \mathcal{C}$, za vsak tajnopis $y \in \mathcal{C}$ obstaja vsaj en ključ K, da je $e_K(x) = y$ in zato velja $|\mathcal{K}| \geq |\mathcal{C}|$.

Za vsako simetrično šifro velja $|\mathcal{C}| \geq |\mathcal{P}|$, saj smo privzeli, da je šifriranje injektivno.

V primeru enakosti (v obeh neenakostih) je Shannon karakteriziral popolno varnost na naslednji način:

Izrek 2. *Naj bo* $(\mathcal{P}, \mathcal{C}, \mathcal{K}, \mathcal{E}, \mathcal{D})$ *simetrična šifra za katero velja* $|\mathcal{K}| = |\mathcal{C}| = |\mathcal{P}|$. *Potem je le-ta popolnoma varna, če in samo, če je vsak ključ uporabljen z enako verjetnostjo* $1/|\mathcal{K}|$ *ter za vsak čistopis* x *in za vsak tajnopis* y *obstaja tak ključ* K, *da je* $e_K(x) = y$.

Dokaz: ($\Longrightarrow$) Ker je $|\mathcal{K}| = |\mathcal{C}|$, sledi, da za vsak čistopis $x \in \mathcal{P}$ in za vsak tajnopis $y \in \mathcal{C}$ obstaja tak ključ K, da je $e_K(x) = y$.

Naj bo $n = |\mathcal{K}|$, $\mathcal{P} = \{x_i \mid 1 \leq i \leq n\}$ in naj za fiksen tajnopis y označimo ključe iz $\mathcal{K}$ tako, da je $e_{K_i}(x_i) = y$ za $i \in [1..n]$. Po Bayesovem izreku velja

$$P(X = x_i/Y = y) = \frac{P(Y = y/x = x_i)\, p_{\mathcal{P}}(x_i)}{p_{\mathcal{C}}(y)}$$

$$= \frac{p_{\mathcal{K}}(K_i)\, p_{\mathcal{P}}(x_i)}{p_{\mathcal{C}}(y)}.$$

Če je šifra popolnoma varna, velja $P(X = x_i/Y = y) = p_{\mathcal{P}}(x_i)$, torej tudi $p_{\mathcal{K}}(K_i) = p_{\mathcal{C}}(y)$, kar pomeni, da je vsak ključ uporabljen z enako verjetnostjo $p_{\mathcal{C}}(y)$ in zato $p_{\mathcal{K}}(K) = 1/|\mathcal{K}|$.

Dokaz obrata poteka na podoben način kot v prejšnjem izreku. ■

Najbolj znana realizacija popolne varnosti je **Vernamov enkratni ščit**, ki ga je leta 1917 patentiral Gilbert Vernam za avtomatizirano šifriranje in odšifriranje telegrafskih sporočil.

Naj bo $\mathcal{P} = \mathcal{C} = \mathcal{K} = (\mathbb{Z}_2)^n$, $n \in \mathbb{N}$,

$$e_K(x) = x \text{ XOR } K,$$

odšifriranje pa je identično šifriranju.

Shannon je prvi po 30-ih letih dokazal, da ta sistem res ne moremo razbiti.

Slabi strani te šifre sta $|\mathcal{K}| \geq |\mathcal{P}|$ in dejstvo, da moramo po vsaki uporabi zamenjati ključ.

Entropija

Doslej nas je zanimala popolna varnost in smo se omejili na primer, kjer uporabimo nov ključ za vsako šifriranje.

Sedaj pa nas zanimata šifriranje vse več in več čistopisa z istim ključem ter verjetnost uspešnega napada z danim tajnopisom in neomejenim časom.

Leta 1948 je Shannon vpeljal v teorijo informacij *entropijo*, tj. matematično mero za informacije oziroma negotovosti in jo izrazil kot funkcijo verjetnostne porazdelitve.

Naj bo X slučajna spremenljivka s končno zalogo vrednosti in porazdelitvijo $p(X)$.

Kakšno informacijo smo pridobili, ko se je zgodil dogodek glede na porazdelitev $p(X)$

oziroma ekvivalentno,

če se dogodek še ni zgodil, kolikšna je negotovost izida?

To količino bomo imenovali **entropija** spremenljivke X in jo označili s $H(X)$.

Primer: metanje kovanca, $p(\text{cifra}) = p(\text{grb}) = 1/2$.

Smiselno je reči, da je entropija enega meta en bit.

Podobno je entropija n-tih metov n, saj lahko rezultat zapišemo z n biti.

Še en primer: slučajna spremenljivka X

$$\begin{pmatrix} x_1 & x_2 & x_3 \\ \frac{1}{2} & \frac{1}{4} & \frac{1}{4} \end{pmatrix}.$$

Najbolj učinkovito zakodiranje izidov je x_1 z 0, x_2 z 10 in x_3 z 11, povprečje pa je

$$\frac{1}{2} \times 1 + \frac{1}{4} \times 2 + \frac{1}{4} \times 2 \;=\; 3/2.$$

Vsak dogodek, ki se zgodi z verjetnostjo 2^{-n}, lahko zakodiramo z n biti.

Posplošitev: dogodek, ki se zgodi z verjetnostjo p, lahko zakodiramo s približno $-\log_2 p$ biti.

Naj bo X slučajna spremenljivka s končno zalogo vrednosti in porazdelitvijo

$$p(X) = \begin{pmatrix} x_1 & x_2 & \dots & x_n \\ p_1 & p_2 & \dots & p_n \end{pmatrix}.$$

Potem **entropijo porazdelitve** $p(X)$ definiramo s

$$H(X) = -\sum_{i=1}^{n} p_i \log_2 p_i = -\sum_{i=1}^{n} p(X = x_i) \log_2 p(X = x_i).$$

Za $p_i = 0$ količina $\log_2 p_i$ ni definirana, zato seštevamo samo po neničelnih p_i (tudi $\lim_{x \to 0} x \log_2 x = 0$).

Lahko bi izbrali drugo logaritemsko bazo, a bi se entropija spremenila le za konstantni faktor.

Če je $p_i = 1/n$ za $1 \leq i \leq n$, potem je $H(X) = \log_2 n$.

Velja $H(X) \geq 0$, enačaj pa velja, če in samo, če je $p_i = 1$ za nek i in $p_j = 0$ za $j \neq i$.

Sedaj pa bomo študirali entropijo različnih komponent simetrične šifre: $H(K)$, $H(P)$, $H(C)$.

Za primer $\mathcal{P} = \{a, b\}$ in $\mathcal{K} = \{K_1, K_2, K_3\}$:

$$p_{\mathcal{P}}(a) = 1/4 \text{ in } p_{\mathcal{P}}(b) = 3/4.$$

$$p_{\mathcal{K}}(K_1) = 1/2 \text{ in } p_{\mathcal{K}}(K_2) = p_{\mathcal{K}}(K_3) = 1/4$$

izračunamo

$$H(P) = -\frac{1}{4}\log_2\frac{1}{4} - \frac{3}{4}\log_2\frac{3}{4} = 2 - \frac{3}{4}\log_2 3 \approx .81 \ .$$

in podobno $H(K) = 1.5$ ter $H(C) \approx 1.85$.

Lastnosti entropije

Realna funkcija f je **(striktno) konkavna** na intervalu I, če za vse (različne) $x, y \in I$ velja

$$f\left(\frac{x+y}{2}\right) \ (>) \ \geq \ \frac{f(x)+f(y)}{2}.$$

Jensenova neenakost: če je f zvezna in striktno konkavna funkcija na intervalu I in $\sum_{i=1}^{n} a_i = 1$ za $a_i > 0$, $1 \leq i \leq n$, potem je

$$f\left(\sum_{i=1}^{n} a_i x_i\right) \geq \sum_{i=1}^{n} a_i f(x_i),$$

enakost pa velja, če in samo, če je $x_1 = x_2 = \cdots = x_n$.

Izrek 3. *$H(X) \leq \log_2 n$, enakost pa velja, če in samo, če je $p_1 = p_2 = \cdots = p_n = 1/n$.*

Izrek 4. *$H(X,Y) \leq H(X) + H(Y)$, enakost pa velja, če in samo, če sta X in Y neodvisni spremenljivki.*

Dokaz izreka 4: Naj bo

$$p(X) = \begin{pmatrix} x_1 & x_2 & \dots & x_m \\ p_1 & p_2 & \dots & p_m \end{pmatrix}, \quad p(Y) = \begin{pmatrix} y_1 & y_2 & \dots & y_n \\ q_1 & q_2 & \dots & q_n \end{pmatrix}$$

in $r_{ij} = p((X = x_i) \cap (Y = y_j))$ za $i \in [1..m]$, $j \in [1..n]$.

Potem za $i \in [1..m]$ in $j \in [1..n]$ velja

$$p_i = \sum_{j=1}^{n} r_{ij} \quad \text{in} \quad q_j = \sum_{i=1}^{m} r_{ij}$$

ter

$$H(X) + H(Y) = -\sum_{i=1}^{m}\sum_{j=1}^{n} r_{ij} \log_2 p_i q_j$$

in

$$H(X,Y) - H(X) - H(Y) = \sum_{i=1}^{m}\sum_{j=1}^{n} r_{ij} \log_2 \frac{p_i q_j}{r_{ij}}$$

$$(\text{Jensen}) \leq \log_2 \sum_{i=1}^{m}\sum_{j=1}^{n} p_i q_j = \log_2 1 = 0.$$

Enakost velja, če in samo, če je $p_i q_j / r_{ij} = c$ za $i \in [1..m]$ in $j \in [1..n]$.

Upoštevajmo še

$$\sum_{j=1}^{n} \sum_{i=1}^{m} r_{ij} = \sum_{j=1}^{n} \sum_{i=1}^{m} p_i q_j = 1$$

in dobimo $c = 1$ oziroma za vse i in j

$$p((X = x_i) \cap (Y = y_j)) = p(X = x_i)\, p(Y = y_j),$$

kar pomeni, da sta spremenljivki X in Y neodvisni. ■

Za slučajni spremenljivki X in Y definiramo **pogojni entropiji**

$$H(X/y) = -\sum_x p(x/y) \log_2 p(x/y)$$

in

$$H(X/Y) = -\sum_y \sum_x p(y)p(x/y) \log_2 p(x/y).$$

Le-ti merita povprečno informacijo spremenljivke X, ki jo odkrijeta y oziroma Y.

Izrek 5. $H(X,Y) = H(Y) + H(X/Y)$.

Dokaz: Po definiciji je $P(X = x_i / Y = y_j) = r_{ij}/q_j$ in $H(Y) + H(X/Y) =$

$$= -\sum_{j=1}^{n}\sum_{i=1}^{m} r_{ij} \log_2 q_j - \sum_{j=1}^{n}\sum_{i=1}^{m} q_j \, r_{ij}/q_j \, \log_2 r_{ij}/q_j \quad \blacksquare$$

Iz izrekov 4 in 5 sledi:

Posledica 6. *$H(X/Y) \leq H(X)$, enakost pa velja, če in samo, če sta X in Y neodvisni spremenljivki.*

Ponarejeni ključi in enotska razdalja

Pogojna verjetnost $H(K/C)$ meri, koliko informacije o ključu je odkrito s tajnopisom.

Izrek 7. *Naj bo $(\mathcal{P}, \mathcal{C}, \mathcal{K}, \mathcal{E}, \mathcal{D})$ simetrična šifra. Potem velja $H(K/C) = H(K) + H(P) - H(C)$.*

Dokaz: Velja $H(K, P, C) = H(C/(K, P)) + H(K, P)$. Ker ključ in čistopis natanko določata tajnopis, je $H(C/K, P) = 0$.

Ker sta P in K neodvisni spremenljivki, dobimo $H(K, P, C) = H(P) + H(K)$ in podobno tudi $H(K, P, C) = H(K, C)$ ter uporabimo še izrek 5. ■

Napadalec privzame, da je čistopis "naravni" jezik (npr. angleščina) in na ta način odpiše mnoge ključe. Vseeno pa lahko ostane še mnogo ključev (med katerimi je le en pravi), ki jih bomo, razen pravega ključa, imenovali **ponarejeni** (angl. spurious).

Naš cilj bo oceniti število ponarejenih ključev.

Naj bo H_L mera povprečne informacije na črko (angl. per letter) v "smiselnem" čistopisu (sledi bolj natančna definicija).

Če so vse črke enako verjetne, je

$$H_L = \log_2 26 \approx 4.70.$$

Kot aproksimacijo *prvega reda* bi lahko vzeli $H(P)$. V primeru angleškega jezika dobimo $H(P) \approx 4.19$.

Tudi zaporedne črke v jeziku niso neodvisne, njihove korelacije pa zmanjšajo entropijo. Za aproksimacijo *drugega reda* bi lahko izračunali entropijo porazdelitve parov črk in potem delili z dve, kajti H_L meri entropijo jezika L na črko.

V splošnem, naj bo P^n slučajna spremenljivka, katere verjetnostna porazdelitev je enaka verjetnostni porazdelitvi n-teric v čistopisu.

Potem je **entropija za naravni jezik** L definirana s

$$H_L = \lim_{n\to\infty} \frac{H(P^n)}{n},$$

odvečnost jezika L pa z

$$R_L = 1 - \frac{H_L}{\log_2 |\mathcal{P}|}.$$

H_L meri entropijo jezika L na črko.

Entropija naključnega jezika je $\log_2 |\mathcal{P}|$.

$R_L \in [0, 1)$ meri kvocient "odvečnih znakov" in je 0 v primeru naključnega jezika.

Za angleški jezik je $H(P^2)/2 \approx 3.90$.

Empirični rezultati kažejo, da je $1.0 \leq H_L \leq 1.5$.

Če ocenimo H_L z 1.25, potem je $R_L \approx .75$, kar pomeni, da je angleščina 75% odvečna

(tj. tekst bi lahko zakodirali le z 1/4 prvotnega teksta).

Podobno kot P^n definiramo še C^n in za $y \in C^n$ še

$$K(y) = \{K \in \mathcal{K} \,|\, \exists x \in P^n,\, p_{\mathcal{P}^n}(x) > 0,\, e_K(x) = y\},$$

tj. $K(y)$ je množica ključev, za katere je y smiselno šifriranje čistopisa dolžine n,

tj. množica verjetnih ključev, za katere je y tajnopis.

Matematično upanje ponarejenih ključev je torej

$$\overline{s}_n = \sum_{y \in \mathcal{C}^n} p(y)(|K(y)| - 1) = \sum_{y \in \mathcal{C}^n} p(y)|K(y)| \; - \; 1.$$

Izrek 8. *Če je* $(\mathcal{P}, \mathcal{C}, \mathcal{K}, \mathcal{E}, \mathcal{D})$ *šifra za katero je* $|\mathcal{C}| = |\mathcal{P}|$ *in so vsi ključi med seboj enakovredni, potem za tajnopis z* n *znaki (*n *je dovolj velik) in za matematično upanje ponarjenih ključev* $\overline{s}_n$ *velja*

$$\overline{s}_n \geq \frac{|\mathcal{K}|}{|\mathcal{P}|^{nR_L}} - 1.$$

Dokaz: Iz izreka 7 sledi

$$H(K/C^n) = H(K) + H(P^n) - H(C^n).$$

Poleg ocene $H(C^n) \leq n \log_2 |\mathcal{C}|$ velja za dovolj velike n tudi ocena $H(P^n) \approx nH_L = n(1 - R_L) \log_2 |\mathcal{P}|$.
Za $|\mathcal{C}| = |\mathcal{P}|$ dobimo

$$H(K/C^n) \geq H(K) - nR_L \log_2 |\mathcal{P}|.$$

Ocenjeno entropijo povežemo še s ponarejenimi ključi

$$H(K/C^n) = \sum_{y \in \mathcal{C}^n} p(y) H(K/y) \leq \sum_{y \in \mathcal{C}^n} p(y) \log_2 |K(y)|$$

$$\leq \log_2 \sum_{y \in \mathcal{C}^n} p(y)|K(y)| = \log_2(\overline{s}_n + 1). \quad \blacksquare$$

Desna stran neenakosti v zadnjem izreku gre z večanjem števila n eksponentno proti 0 (to ni limita, števila $|\mathcal{K}|$, $|\mathcal{P}|$ in R_L so fiksna, število $|\mathcal{K}|$ pa je običajno veliko v primerjavi s $|\mathcal{P}|^{R_L} > 1$).

Enotska razdalja simetrične šifre je tako število n, označeno z n_0, za katerega postane matematično upanje ponarejenih ključev nič, tj. povprečna dolžina tajnopisa, ki jo napadalec potrebuje za računanje ključa pri neomejenem času.

Velja $$n_0 \approx \frac{\log_2 |\mathcal{K}|}{R_L \log_2 |\mathcal{P}|}.$$

V primeru zamenjalnega tajnopisa sta $|\mathcal{P}| = 26$ in $|\mathcal{K}| = 26!$. Če vzamemo $R_L = .75$, potem je enotska razdalja

$$n_0 \approx \frac{88.4}{.75 \times 4.7} \approx 25 .$$

Produktne šifre

Še ena Shannonova ideja v članku iz leta 1949 igra danes pomembno vlogo, predvsem pri simetričnih šifrah.

Zanimali nas bodo šifre, za katere $\mathcal{C} = \mathcal{P}$,

tj. **endomorfne šifre**.

Naj bosta $S_i = (\mathcal{P}, \mathcal{P}, \mathcal{K}_i, \mathcal{E}_i, \mathcal{D}_i)$, $i = 1, 2$, endomorfni simetrični šifri. Potem je **produkt** sistemov S_1 in S_2, označen s $S_1 \times S_2$, definiran s

$$(\mathcal{P}, \mathcal{P}, \mathcal{K}_1 \times \mathcal{K}_2, \mathcal{E}, \mathcal{D})$$

ter

$$e_{(K_1,K_2)}(x) = e_{K_2}(e_{K_1}(x))$$

in

$$d_{(K_1,K_2)}(y) = d_{K_1}(e_{K_2}(y)).$$

Njegova verjetnostna porazdelitev pa naj bo

$$p_{\mathcal{K}}(K_1, K_2) = p_{\mathcal{K}_1}(K_1) \times p_{\mathcal{K}_2}(K_2),$$

tj. ključa K_1 in K_2 izberemo neodvisno.

Če sta M in S zaporedoma multiplikativni tajnopis in zamični tajnopis, potem je $M \times S$ afin tajnopis. Malce težje je pokazati, da je tudi tajnopis $S \times M$ afin tajnopis. Ta dva tajnopisa torej **komutirata**.

Vsi tajnopisi ne komutirajo, zato pa je produkt asociativna operacija:

$$(S_1 \times S_2) \times S_3 = S_1 \times (S_2 \times S_3).$$

Če je $(S \times S =)\ S^2 = S$, pravimo, da je sistem **idempotenten**.

Zamični, zamenjalni, afin, Hillov, Vigenèrov in permutacijski tajnopisi so vsi idempotentni.

Če simetrična šifra ni idempotentna, potem se zna zgoditi, da z njeno iteracijo za večkrat povečamo varnost. Na tem so zasnovani **DES** in mnoge druge simetrične šifre.

Če sta simetrični šifri S_1 in S_2 idempotentni in obenem še komutirata, potem se ni težko prepričati, da je tudi produkt $S_1 \times S_2$ idempotentna simetrična šifra.

3. poglavje

Simetrični kriptosistemi

- Bločne šifre, nekaj zgodovine, DES, AES
- Iterativne šifre, zmenjalno-permutacijske mreže
- Produktna šifra in Fiestelova šifra
- Opis šifer DES in AES
- Načini delovanja (ECB, CBC, CFB, OFB) in MAC
- Napadi in velika števila
- 3-DES, DESX in druge simetrične bločne šifre

Bločne šifre

Bločna šifra je simetrična šifra, ki razdeli čistopis na bloke fiksne dolžine (npr. 128 bitov), in šifrira vsak blok posamično (kontrast: *tekoča šifra* zašifrira čistopis po znakih – ponavadi celo po bitih).

Najmodernejše bločne šifre so **produktne šifre**, ki smo jih spoznali v prejšnjem poglavju: komponiranje več enostavnih operacij, katere (vsaka posebej) niso dovolj varne, z namenom, da povečamo varnost: *transpozicije, ekskluzivni ali (XOR), tabele, linearne transformacije, aritmetične operacije, modularno množenje, enostavne substitucije.*

Primeri bločnih produktnih šifer: DES, AES, IDEA.

Nekatere želene lastnosti bločnih šifer

Varnost:

- **razpršitev**: vsak bit tajnopisa naj bo odvisen od vseh bitov čistopisa.
- **zmeda**: zveza med ključem ter biti tajnopisa naj bo zapletena,
- **velikost ključev**: mora biti majhna, toda dovolj velika da prepreči požrešno iskanje ključa.

Učinkovitost

- hitro šifriranje in odšifriranje,
- enostavnost (za lažjo implementacijo in analizo),
- primernost za hardware ali software.

Kratka zgodovina bločnih šifer DES in AES

Konec 1960-ih: IBM – Feistelova šifra in LUCIFER.

1972: NBS (sedaj NIST) izbira simetrično šifro za zaščito računalniških podatkov.

1974: IBM razvije DES, 1975: NSA ga "popravi".

1977: DES sprejet kot US Federal Information Processing Standard (FIPS 46).

1981: DES sprejet kot US bančni standard (ANSI X3.92).

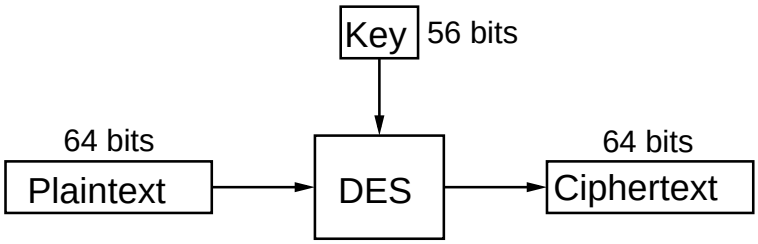
1997: AES (Advanced Encryption Standard) izbor

1999: izbranih 5 finalistov za AES

National Security Agency (NSA)

- www.nsa.gov
- ustanovljena leta 1952,
- neznana sredstva in število zaposlenih (čez 100.000?)
- Signals Intelligence (SIGINT):
  pridobiva tuje informacije.
- Information Systems Security (INFOSEC):
  ščiti vse občutljive (classified) informacije,
  ki jih hrani ali pošilja vlada ZDA,
- zelo vplivna pri določanju izvoznih regulacij ZDA za kriptografske produkte (še posebej šifriranje).

Data Encryption Standard (DES)



Ideja za DES je bila zasnovana pri IBM-u v 60-ih letih (uporabili so koncept Claude Shannona imenovan *Lucifer*).

NSA je zreducirala dolžino ključev s 128 bitov na 56.

V sredini 70-ih let je postal prvi komercijalni algoritem, ki je bil objavljen z vsemi podrobnostmi (FIPS 46-2).

Advanced Encryption Standard

AES je ime za nov FIPS-ov simetrični (bločni) kriptosistem, ki bo nadomestil DES.

Leta 2000 je zanj *National Institute of Standards and Technology (NIST)* izbral belgijsko bločno šifro **Rijndael**.

Dolžina *ključev* oziroma blokov je 128, 192 ali 256

Uporabljala pa ga tudi ameriška vlada, glej

`http://csrc.nist.gov/encryption/aes/round2/r2report.pdf`.

Običajno uporabljamo **iterativne šifre**.

Tipični opis:

- krožna funkcija,
- razpored ključev,
- šifriranje skozi N_r podobnih krogov.

Naj bo K naključni binarni ključ določene dolžine.

K uporabimo za konstrukcijo podključev za vsak krog s pomočjo *javno* znanega algoritma.

Imenujemo jih **krožni ključi:** $K^1, \ldots, K^{N_r}$.

Seznamu krožnih ključev $(K^1, \ldots, K^{N_r})$ pa pravimo **razpored ključev**.

Krožna funkcija g ima dva argumenta:

(i) krožni ključ (K^r) in (ii) *tekoče stanje* (w^{r-1}).

Naslednje stanje je definirano z $w^r = g(w^{r-1}, K^r)$.

Začetno stanje, w_0, naj bo čistopis x.

Potem za tajnopis, y, vzamemo stanje po N_r krogih:

$$y = g(g(\ldots g(g(x, K^1), K^2) \ldots, K^{N_r-1})K^{N_r}).$$

Da je odšifriranje možno, mora biti funkcija g injektivna za vsak fiksen ključ K_i, tj. $\exists\ g^{-1}$, da je:

$$g^{-1}(g(w, K), K) = w, \qquad \text{za vse } w \text{ in } K.$$

Odšifriranje opravljeno po naslednjem postopku:

$$x = g^{-1}(g^{-1}(\ldots g^{-1}(g^{-1}(y, K^{N_r}), K^{N_r-1}) \ldots, K^2)K^1).$$

Zamenjalno-permutacijske mreže

(angl. *substitution-permutation network* – (**SPN**)).

Čistopis $\mathcal{P}$ in tajnopis $\mathcal{C}$ so binarni vektorji dolžine ℓm, $\ell, m \in \mathbb{N}$ (tj. ℓm je dolžina bloka).

SPN je zgrajen iz dveh komponent (zamenjave in permutacije):

$$\begin{aligned} \pi_S &: \{0,1\}^\ell \longrightarrow \{0,1\}^\ell, \\ \pi_P &: \{0,\ldots,\ell m\} \longrightarrow \{0,\ldots,\ell m\}. \end{aligned}$$

Permutacijo π_S imenujemo S-**škatla** in z njo zamenjamo ℓ bitov z drugimi ℓ biti.

Permutacija π_P pa permutira ℓm bitov.

Naj bo $x = (x_1, \ldots, x_{\ell m})$ binarno zaporedje, ki ga lahko smatramo za spoj m ℓ-bitnih podzaporedij označenih z $x_{(1)}, \ldots, x_{(m)}$.

SPN ima N_r krogov, v vsakem (razen zadnjem, ki je bistveno drugačen) opravimo m zamenjav z π_S in nato uporabimo še π_P. Pred vsako zamenjavo vključimo krožni ključ z XOR operacijo.

SPN šifra

$\ell, m, N_r \in \mathbb{N}$, π_S in π_P permutaciji, $\mathcal{P} = \mathcal{C} = \{0,1\}^{\ell m}$ in $\mathcal{K} \subseteq (\{0,1\}^{\ell m})^{N_r+1}$, ki se sestoji iz vseh možnih razporedov ključev izpeljanih iz ključa K z uporabo algoritma za generiranje razporeda ključev.
Šifriramo z algoritmom SPN.

Alg. : *SPN*$(x, \pi_S, \pi_P, (K^1, \ldots, K^{N_r+1}))$

$w^0 := x$

for $r := 1$ **to** $N_r - 1$ **do** *(krožno mešanje ključev)*

$u^r := w^{r-1} \oplus K^r$

for $i := 1$ **to** m **do** $v^r_{(i)} := \pi_S(u^r_{(i)})$

$w^r := (v^r_{\pi_P(1)}, \ldots, v^r_{\pi_P(\ell m)})$

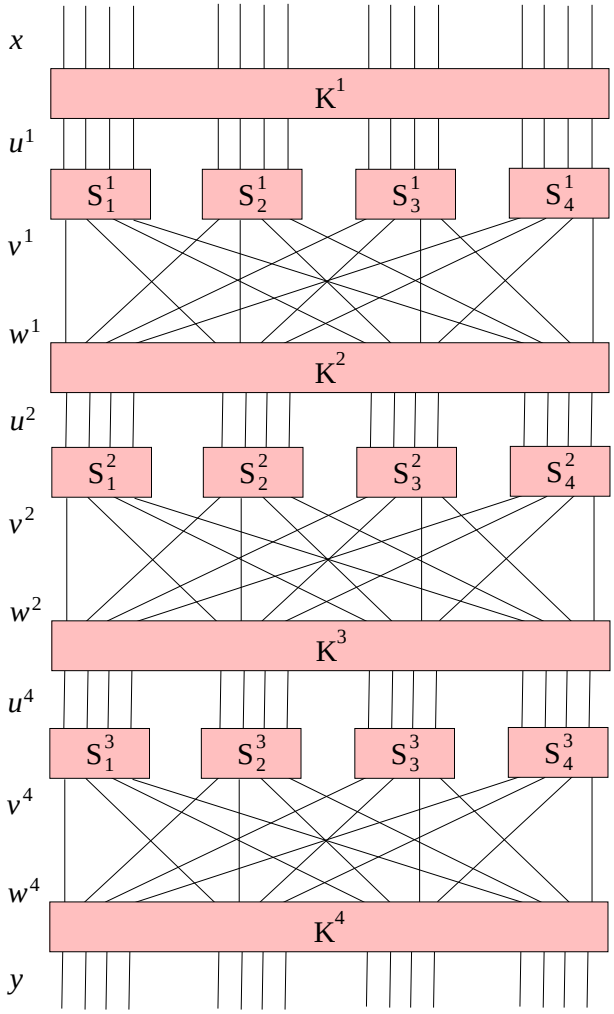
(zadnji krog)

$u^{N_r} := w^{N_r-1} \oplus K^{N_r}$

for $i := 1$ **to** m **do** $v^{N_r}_{(i)} := \pi_S(u^{N_r+1}_{(i)})$

$y := v^{N_r} \oplus K^{N_r+1}$

output (y)


x
K^1
u^1
S_1^1
S_2^1
S_3^1
S_4^1
v^1
w^1
K^2
u^2
S_1^2
S_2^2
S_3^2
S_4^2
v^2
w^2
K^3
u^4
S_1^3
S_2^3
S_3^3
S_4^3
v^4
w^4
K^4
y

Primer: naj bo $\ell = m = N_r = 4$, permutaciji π_S in π_P pa podani s tabelami:

z	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
$\pi_S(z)$	E	4	D	1	2	F	B	8	3	A	6	C	5	9	0	7

ter

z	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
$\pi_P(z)$	1	5	9	13	2	6	10	14	3	7	11	15	4	8	12	16

Naj bo ključ $K = (k_1, \ldots, k_{32}) \in \{0,1\}^{32}$ definiran z

$$K = \texttt{0011 1010 1001 0100 1101 0110 0011 1111},$$

sedaj pa izberimo še razpored ključev tako, da je za $1 \leq r \leq 5$, krožni ključ K^r izbran kot 16 zaporednih bitov ključa K z začetkom pri k_{4r-3}:

$$\begin{aligned} K^1 &= 0011\ 1010\ 1001\ 0100 \\ K^2 &= 1010\ 1001\ 0100\ 1101 \\ K^3 &= 1001\ 0100\ 1101\ 0110 \\ K^4 &= 0100\ 1101\ 0110\ 0011 \\ K^5 &= 1101\ 0110\ 0011\ 1111 \end{aligned}$$

Potem šifriranje čistopisa

$$x = 0010\ 0110\ 1011\ 0111$$

poteka v naslednjem vrstnem redu.

$$
\begin{aligned}
w^0 &= 0010\ 0110\ 1011\ 0111, & K^1 &= 0011\ 1010\ 1001\ 0100\\
u^1 &= 0001\ 1100\ 0010\ 0011, & v^1 &= 0100\ 0101\ 1101\ 0001\\
w^1 &= 0010\ 1110\ 0000\ 0111, & K^2 &= 1010\ 1001\ 0100\ 1101\\
u^2 &= 1000\ 0111\ 0100\ 1010, & v^2 &= 0011\ 1000\ 0010\ 0110\\
w^2 &= 0100\ 0001\ 1011\ 1000, & K^3 &= 1001\ 0100\ 1101\ 0110\\
u^3 &= 1101\ 0101\ 0110\ 1110, & v^3 &= 1001\ 1111\ 1011\ 0000\\
w^3 &= 1110\ 0100\ 0110\ 1110, & K^4 &= 0100\ 1101\ 0110\ 0011\\
u^4 &= 1010\ 1001\ 0000\ 1101, & v^4 &= 0110\ 1010\ 1110\ 1001\\
K^5 &= 1101\ 0110\ 0011\ 1111, & y &= 1011\ 1100\ 1101\ 0110
\end{aligned}
$$

Možno so številne varijacije SPN šifer.

Na primer, namesto ene S-škatle lahko uporabimo različne škatle. To lahko vidimo pri DES-u, ki uporabi 8 različnih škatel.

Zopet druga možnost je uporabiti obrnljive linearne transformacije, kot zamenjavo za permutacije ali pa samo dodatek. Tak primer je AES.

Feistelova šifra

Feistelova šifra: r krogov (rund)

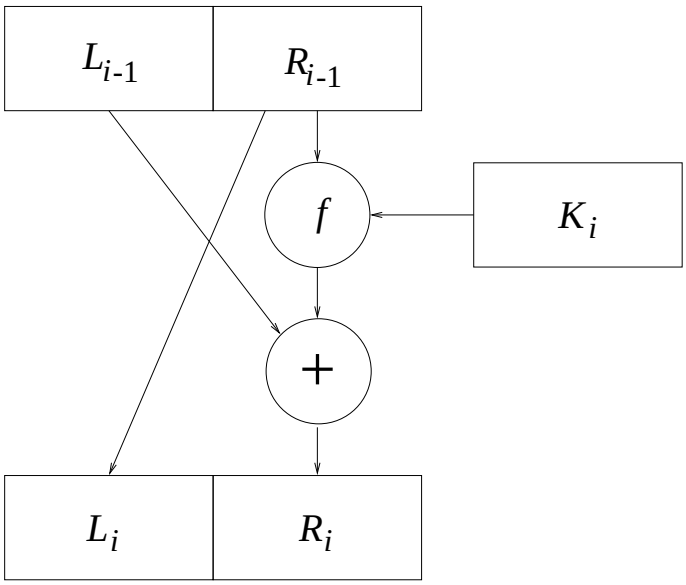
$$(L_{i-1}, R_{i-1}) \xrightarrow{K_i} (L_i, R_i).$$

kjer je $L_i = R_{i-1}$ in $R_i = L_{i-1} \oplus f(R_{i-1}, K_i)$,
in smo podključe K_i dobili iz osnovnega ključa K.

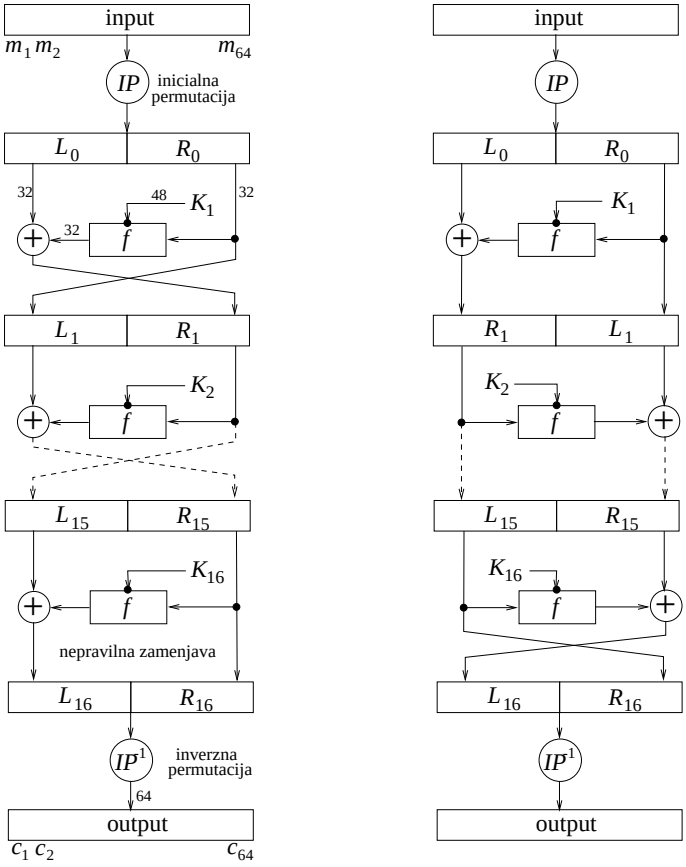
Končamo z (R_r, L_r) (in ne z (L_r, R_r)), zato je šifriranje enako odšifriranju, le da ključe uporabimo v obratnem vrstnem redu.

Funkcija f je lahko produktna šifra in ni nujno obrnljiva.

En krog



Opis šifre DES



DES-ove konstante

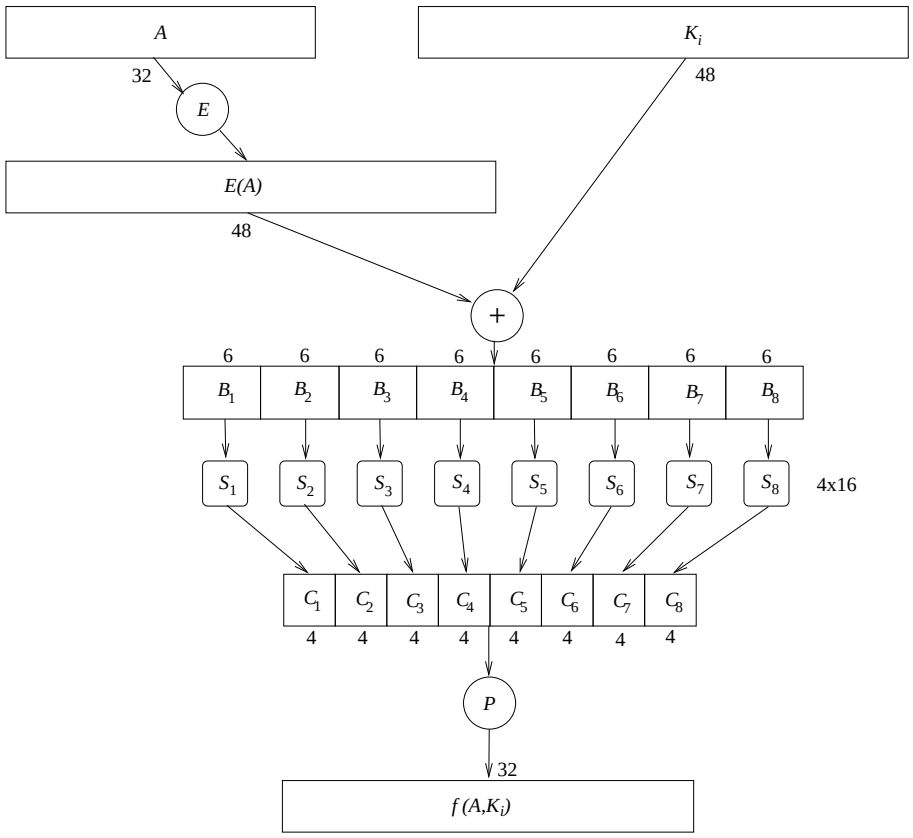
začetna in končna permutacija: IP, IP^{-1}

razširitev: E (nekatere bite ponovimo), permutacija P

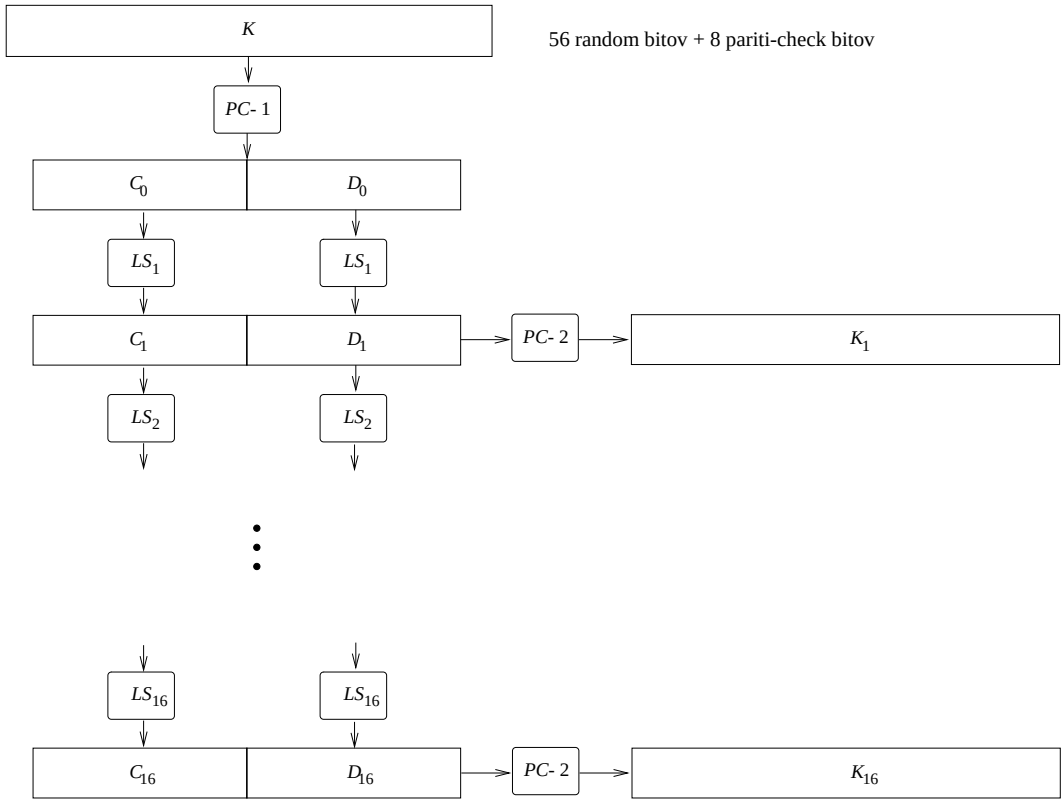
S-škatle: $S_1, S_2, \ldots, S_8$
(tabele: 4×16, z elementi $0 - 15$)

permutacije za gen. podključev: $PC-1$, $PC-2$

DES-ova funkcija



Računanje DES-ovih ključev



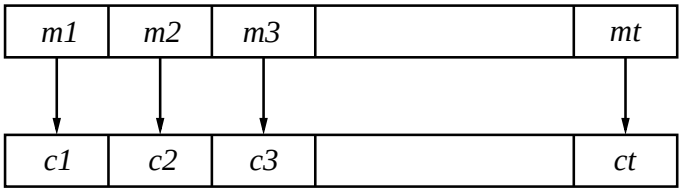
20 let je DES predstavljal delovnega konja kriptografije (bločnih šifer).

- do leta 1991 je NBS sprejel 45 hardwarskih implementacij za DES
- geslo (PIN) za bankomat (ATM)
- ZDA (Dept. of Energy, Justice Dept., Federal Reserve System)

Načini delovanja simetričnih šifer

– electronic codebook mode **(ECB)**
– cipher block chaining mode **(CBC)**
– cipher feedback mode **(CFB)**
– output feedback mode **(OFB)**

Pri **ECB** šifriramo zaporedoma blok po blok:
$c = c_1, c_2, \ldots, c_t$, kjer je $c_i = E_k(m_i)$.



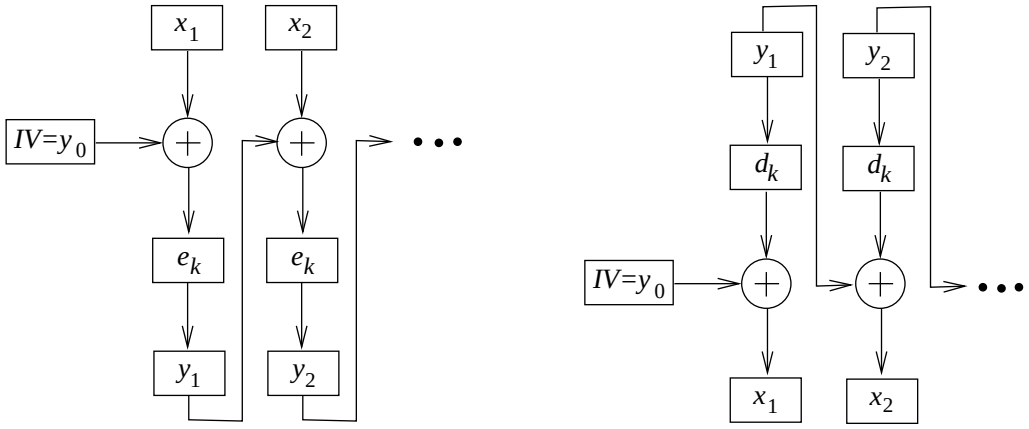
Odšifriranje: $m_i = D_k(c_i)$, $i = 1, 2, \ldots, t$.

Slabost: identični bloki čistopisa se (pri istem ključu) zašifrirajo v identične bloke tajnopisa.

Cipher Block Chaining mode – CBC

čistopis/tajnopis: 64 bitni bloki $x_1, x_2, \ldots / y_1, y_2, \ldots$

Šifriranje: $y_0 := \mathrm{IV}$, $y_i := e_K(y_{i-1} \oplus x_i)$ za $i \geq 1$.



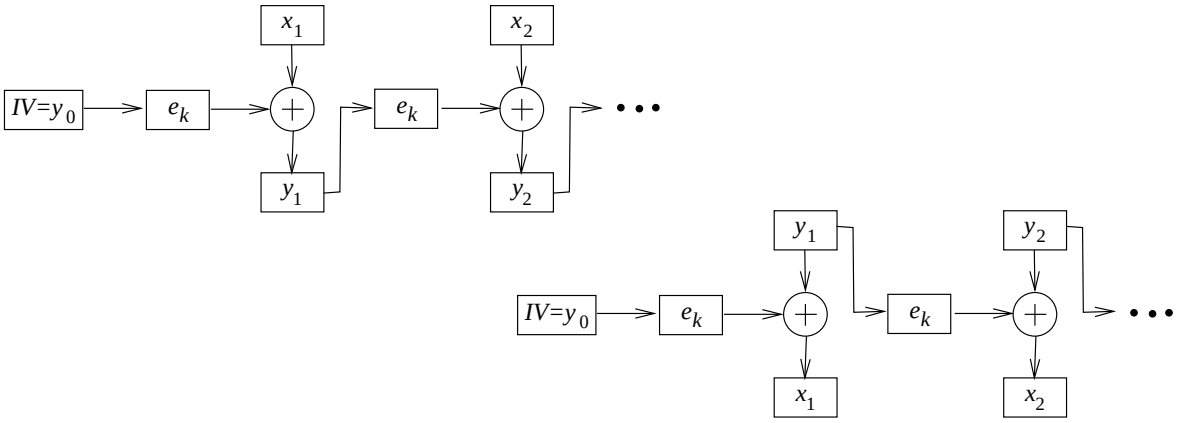
Odšifriranje: $y_0 := \mathrm{IV}$, $x_i := y_{i-1} \oplus d_K(y_i)$ za $i \geq 1$.

Identičena čistopisa z različnimi IV dasta različen tajnopis. Eno-bitna napaka pri tajnopisu pokvari le odšifriranje dveh blokov.

Cipher Feedback mode – CFB

čistopis/tajnopis: 64 bitni bloki $x_1, x_2, \ldots / y_1, y_2, \ldots$

$y_0 := \mathrm{IV}$, šifriranje: $z_i := e_K(y_{i-1})$, $y_i := y_{i-1} \oplus x_i$, $i \geq 1$.



Odšifriranje ($y_0 := \mathrm{IV}$, $x_0 = d_K(\mathrm{IV})$):

$$z_i := d_K(x_{i-1}) \text{ in } x_i := y_i \oplus z_i \quad \text{za} \quad i \geq 1.$$

CFB se uporablja za preverjanje celovitosti sporočila (angl. message authentication code - MAC).

Output Feedback mode – OFB

čistopis/tajnopis: 64 bitni bloki $x_1, x_2, \ldots / y_1, y_2, \ldots$

Inicializacija: $z_0 := \mathrm{IV}$, šifriranje:

$$z_i := e_K(z_{i-1}) \quad \text{in} \quad y_i := x_i \oplus z_i \quad \text{za} \quad i \geq 1.$$

Odšifriranje:$(z_0 := \mathrm{IV})$

$$z_i := e_K(z_{i-1}) \quad \text{in} \quad x_i := y_i \oplus z_i \quad \text{za} \quad i \geq 1.$$

OFB se uporablja za satelitske prenose.

Napadi na šifro DES

Požrešni napad: preverimo vseh 2^{56} ključev.

Leta 1993 Michael J. Wiener, Bell-Northern Research, Kanada, predstavi učinkovito iskanje DES ključa:

- **diferenčna kriptoanaliza** z 2^{47} izbranimi čistopisi (Biham in Shamir 1989)
  – je učinkovita tudi na nekaterih drugih bločnih šifrah,
- **linearna kriptoanaliza** z 2^{47} poznanimi čistopisi (Matsui 1993):

Slednja napada sta statistična, saj potrebujeta velike količine čistopisa in ustreznega tajnopisa, da določita ključ. Pred leti sta bila napada zanimiva le teoretično.

Wienerjev cilj je bil precizna ocena časa in denarja potrebnega za graditev čipov za iskanje DES ključa.

Požrešna metoda na prostor ključev: 2^{56} korakov je zlahka paralelizirana.

Dan je par čistopis-tajnopis (P, C) ter začetni ključ K. Registri za vsako iteracijo so ločeni, tako da je vse skupaj podobno tekočemu traku:

- hitrost 50 MHz
- cena \$10.50 na čip
- 50 milijonov ključev na sekundo
- skupaj: \$100 tisoč, 5760 čipov, rabi 35 ur

Pri linearni kriptoanalizi hranjenje parov zavzame 131,000 Gbytov. Implementirano leta 1993: 10 dni na 12 mašinah.

Po odkritju diferenčne kriptoanalize je
Don Coppersmith priznal, da je IBM v resnici
poznal ta napad (ne pa tudi linearno kriptoanalizo)
že ko so razvijali DES:

"Po posvetovanju z NSA, smo se zavedali, da utegne objava kriterijev načrtovanja odkriti tehniko kriptoanalize. To je močno sredstvo, ki se ga da uporabiti proti mnogim tajnopisom.
To bi zmanjšalo prednost ZDA pred drugimi na področju kriptografije."

Novejši rezultati napadov

DES izivi pri RSA Security (3 poznani PT/CT pari):

T h e u n k n	o w n m e s s	a g e i s :	? ? ? ? ? ? ? ?

junij 1997: razbito z internetnim iskanjem (3m).

julij 1998: razbito v treh dneh z DeepCrack mašino (1800 čipov; $250,000).

jan. 1999: razbita v 22 h, 15 min (DeepCrack + porazdeljena.mreža).

V teku (porazdeljena.mreža): RC5 – 64-bitni izziv:

- pričeli konec 1997; trenutna hitrost: 2^{36} ključev/sec (2^{25} secs/leto; pričakovani čas: ≤ 8 let).

Implementacijski napadi na DES

Napadi s pomočjo diferenčne analize porabe moči
(angl. differential power analysis (**DPA**) attacks):

- Kocher, Jaffe, Jun 1999,
- procesorjeva poraba moči je odvisna od instrukcij,
- merimo porabo moči inštrukcij, ki se izvedejo v 16-ih krogih DES-a
- ≈1000 tajnopisa zadoščajo za odkritje tajnega ključa.

Napadi s pomočjo diferenčne analize napak
(angl. differential fault analysis (**DFA**) attacks):

- Biham, Shamir 1997.
- napad: zberi naključne napake v 16-ih krogih DES-a.
- ≈200 napačnih odšifriranj zadošča za razkritje tajnega ključa.

Vse o napadih je veljalo za ECB način.

Isti čipe se da uporabiti tudi za druge načine, cena in čas pa se nekoliko povečata. Recimo po Wienerju za CBC način rabimo \$1 milijon in 4 ure.

Varnost DES-a lahko enostavno povečamo,
če uporabimo **3-DES** (zakaj ne 2-DES?).

$$\begin{aligned}&\mathrm{DES_E}(\mathrm{P},\mathrm{K}_1)\longrightarrow \mathrm{DES_D}(\mathrm{DES_E}(\mathrm{P},\mathrm{K}_1),\mathrm{K}_2)\\ &\longrightarrow \mathrm{DES_E}(\mathrm{DES_D}(\mathrm{DES_E}(\mathrm{P},\mathrm{K}_1),\mathrm{K}_2),\mathrm{K}_3)\end{aligned}$$

Za $K_1 = K_2 = K_3$ dobimo običajni DES.

Običajno pa zamenjamo K_3 s K_1 in dobimo približno za faktor 10^{13} močnejši sistem.

Kako veliko je VELIKO?

sekund v enem letu		$\approx 3 \times 10^7$
(živimo "le" 2-3 milijarde sekund)		
starost našega sončnega sistema (v letih)		$\approx 6 \times 10^9$
urinih ciklov na leto (200 MHz)		$\approx 6.4 \times 10^{15}$
01-zaporedij dolžine 64	$\approx 2^{64}$	$\approx 1.8 \times 10^{19}$
01-zaporedij dolžine 128	$\approx 2^{128}$	$\approx 3.4 \times 10^{38}$
01-zaporedij dolžine 256	$\approx 2^{256}$	$\approx 1.2 \times 10^{77}$
75 številčnih praštevil		$\approx 5.2 \times 10^{72}$
elektronov v vsem vesolju		$\approx 8.37 \times 10^{77}$

mega (M)	giga (G)	tera (T)	peta (P)	exa (E)
10^6	10^9	10^{12}	10^{15}	10^{18}

Računska moč

za naše potrebe bomo privzeli, da se smatra:

- 2^{40} operacij za *lahko*,
- 2^{56} operacij za *dosegljivo*,
- 2^{64} operacij za *komaj da dosegljivo*,
- 2^{80} operacij za *nedosegljivo*,
- 2^{128} operacij za *popolnoma nedosegljivo*.

3-DES je trikrat počasnejši od DES-a.

To je pogosto nesprejemljivo, zato je leta 1984 Ron Rivest predlagal **DESX**:

$$\boxed{\mathrm{DESX}_{\mathrm{k.k1.k2}}(\mathrm{x}) = \mathrm{k2} \oplus \mathrm{DES}_{\mathrm{k}}(\mathrm{k1} \oplus \mathrm{x}).}$$

DESX ključ $K = k.k1.k2$ ima

$$56 + 64 + 64 = 184 \text{ bitov.}$$

DESX trik onemogoči preizkušanje vseh mogočih ključev (glej P. Rogaway, 1996). Sedaj rabimo več kot 2^{60} izbranega čistopisa.

Hitrost

Preneel, Rijmen, Bosselaers 1997.

Softwarski časi za implementacijo na 90MHz Pentiumu.

šifra	velikost ključa (biti)	hitrost
DES	56	10 Gbits/sec (ASIC chip)
DES	56	16.9 Mbits/sec
3DES	128	6.2 Mbits/sec
RC5-32/12	128	38.1 Mbits/sec
Arcfour	variable	110 Mbits/sec