Symetrické šifry 2 crypto

Challenge status: Available
Challenge flags: 4

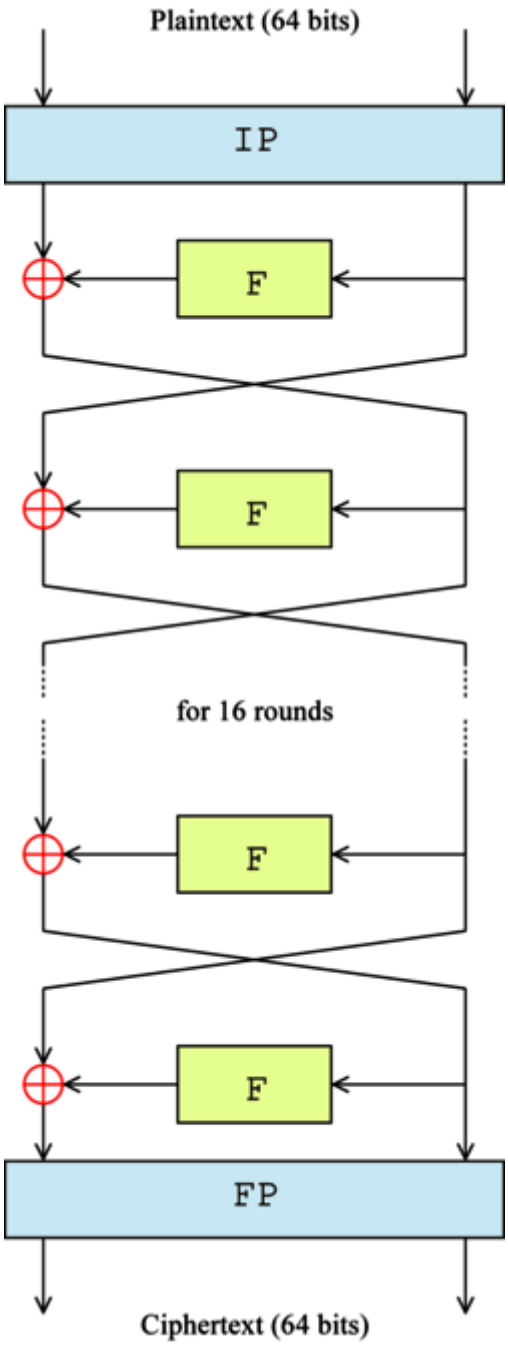
232

Ponořte se do další úlohy symetrické kryptografie a prozkoumejte detaily funkce F ve Feistelově šifře, která je klíčovou součástí algoritmu DES. Postupně si vyzkoušíte všechny čtyři části funkce, přesněji expanzi, mixování klíče, substituci pomocí S-boxů a permutaci.

Theory Flags Handbook Solves

Úvod

Minule jsme se naučili, jak funguje iniciální a finální permutace a Feistelova šifra v algoritmu DES. Vstupní data se za IP rozdělí na dvě poloviny a prochází Feistelovou šifrou.



Feistelova šifra v algoritmu DES

Dnes se podíváme na funkci F ve Feistelově šifře.

Funkce F

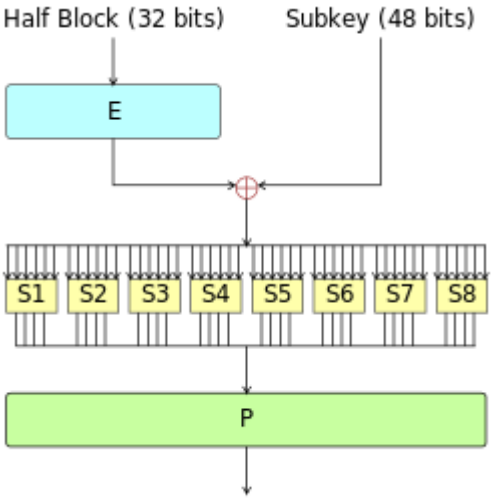
V matematice jste se už určitě setkali s funkcemi, například `f(x) = 2x`. `f` je jméno funkce a `(x)` značí, že je závislá na proměnné x. Za rovná se je předpis funkce `2x`. To už všichni dobře znáte a není to nic nového. Funkce ale může být mnohem více abstraktní a složitější. Jako například funkce F v algoritmu DES. Opět to je funkce, akorát je závislá na vstupních datech (32 bitech) a sub klíči (48 bitů).

Proč mají vstupní data 32 bitů už víte, jedná se o polovinu bloku. A proč sub klíč má 48 bitů si také určitě pamatujete, protože se z

původních 56 bitů vytvoří pro každý krok ve Feistově šifře odlišný 48bitový sub klíč.

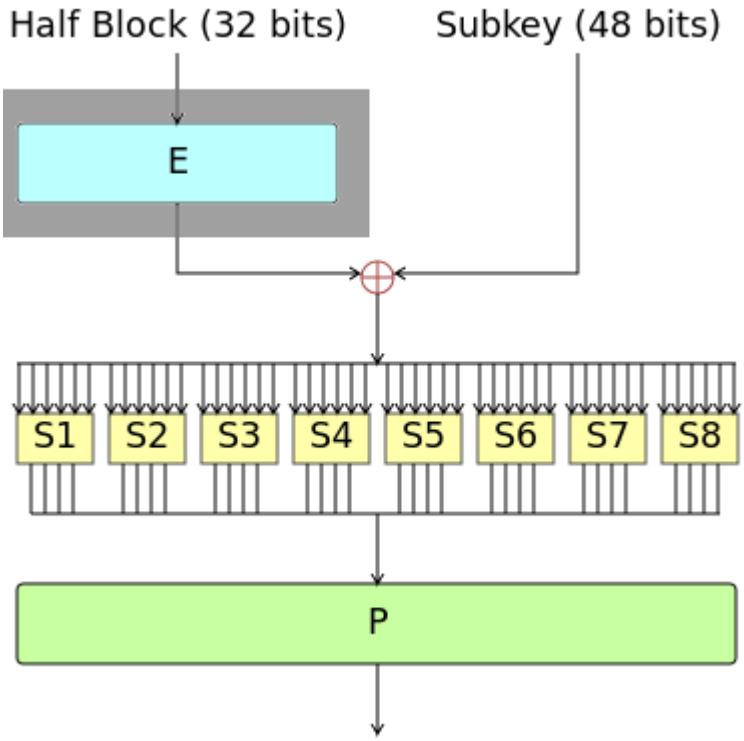
Ano, je to podstatně velký skok z Caesarovi nebo XOR šifry :) Generaci sub klíčů vynecháme, ale pojďme se podívat, jak funguje funkce F, která se dá rozdělit do čtyř kroků:

- Expanze (E)
- Mixování klíče (XOR)
- Substituce (S-boxy)
- Permutace (P)



Expanze (E)

Z původních 32 bitů (poloviny bloku) vytvoří 48 bitů.

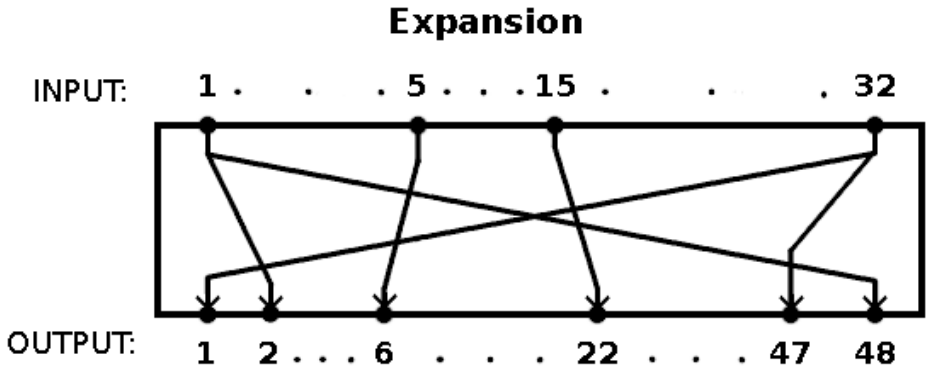


Funguje stejně, jako tabulka IP a FP

Používá se tabulka, která udává, ze kterého bitu vstupu je brán bit výstupu. První buňka (levá, horní) značí první bit výstupu a obsah buňky udává, z kolikátého bitu vstupu se berou data.

32	1	2	3	4	5
4	5	6	7	8	9
8	9	10	11	12	13
12	13	14	15	16	17
16	17	18	19	20	21
20	21	22	23	24	25
24	25	26	27	28	29
28	29	30	31	32	1

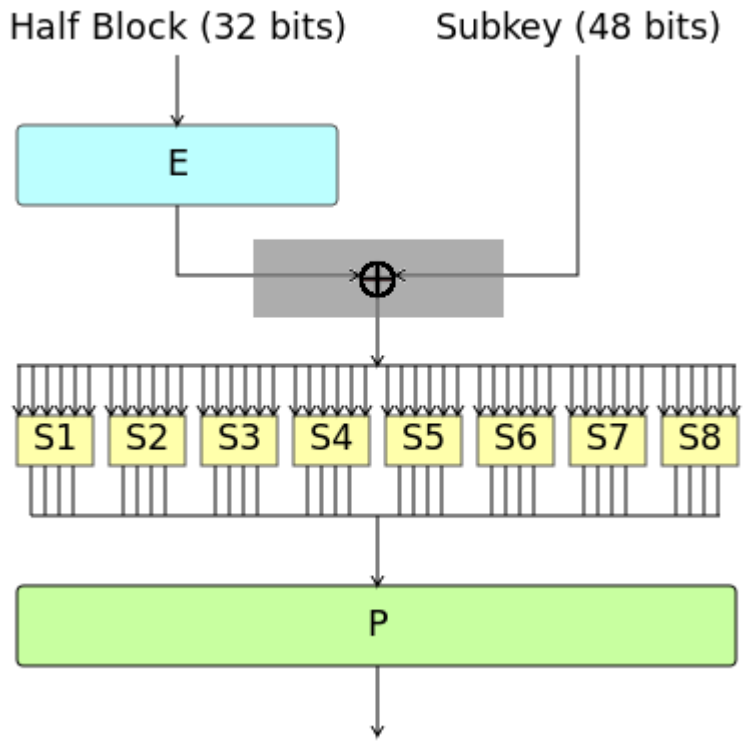
Tabulku čtěte po řádcích. První bit výstupu se bere z 32. bitu vstupu, druhý bit výstupu se bere z 1. bitu vstupu... až poslední 48. bit výstupu se bere z 1. bitu vstupu. Asi jste si všimnuli, že se některé vstupní bity používají vícekrát, ano právě opakováním některých vstupních bitů se z původních 32 bitů vytvoří 48.



Znázornění expanze

Mixování klíče (XOR)

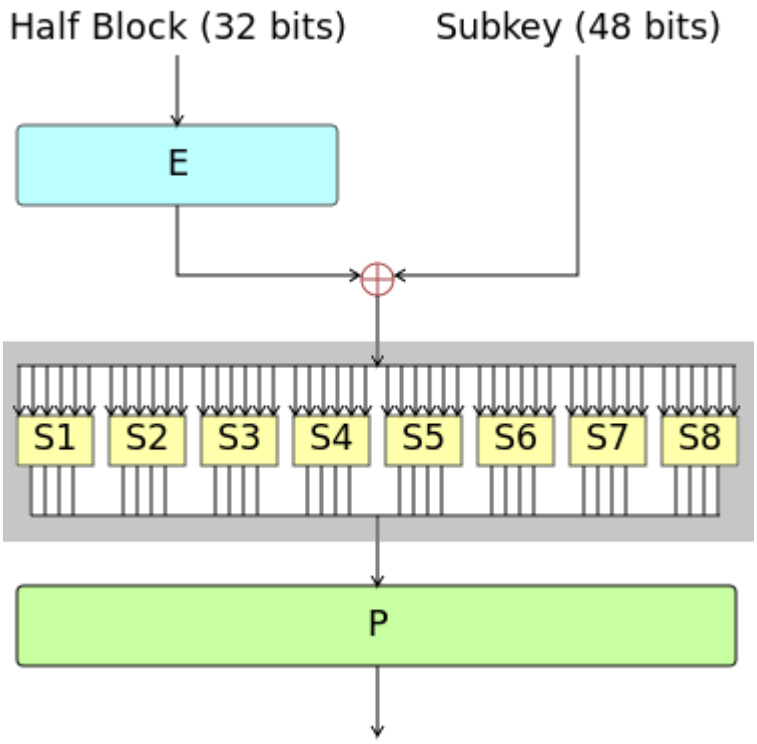
Zde je to jednoduché, XORne se 48 bitů expandované poloviny bloku s 48bitovým sub klíčem.



> *Znak ⊕ značí v nákresu XOR.*

Substituce (S-boxy)

Substitution-**box**



Zde se výsledek předchozího kroku rozdělí po šesti bitech a každá šestice putuje do jednoho S-boxu, první S-box je označen S1, druhý S2 atd...

Jednotlivé S-boxy se řídí tabulkou a podle ní z šestic vytvoří čtveřice. Substituují (nahrazují) za původních 6 bitů 4.

Zde je tabulka S-boxu 1 (S1). Řádek se vybírá pomocí dvou vnějších bitů a sloupec pomocí čtyř vnitřních bitů. Výsledek je v desítkové soustavě

S1	x0000x	x0001x	x0010x	x0011x	x0100x	x0101x	x0110x
0yyyy0	14	4	13	1	2	15	11
0yyyy1	0	15	7	4	14	2	13
1yyyy0	4	1	14	8	13	6	2
1yyyy1	15	12	8	2	4	9	1

Například číslo **1**0011**0** se rozdělí na dva **vnější** bity **10** a čtyři vnitřní bity 0011. V tabulce to je třetí řádek a čtvrtý sloupec. Výsledek je v desítkové soustavě 8, po převedení do binární je 1000.

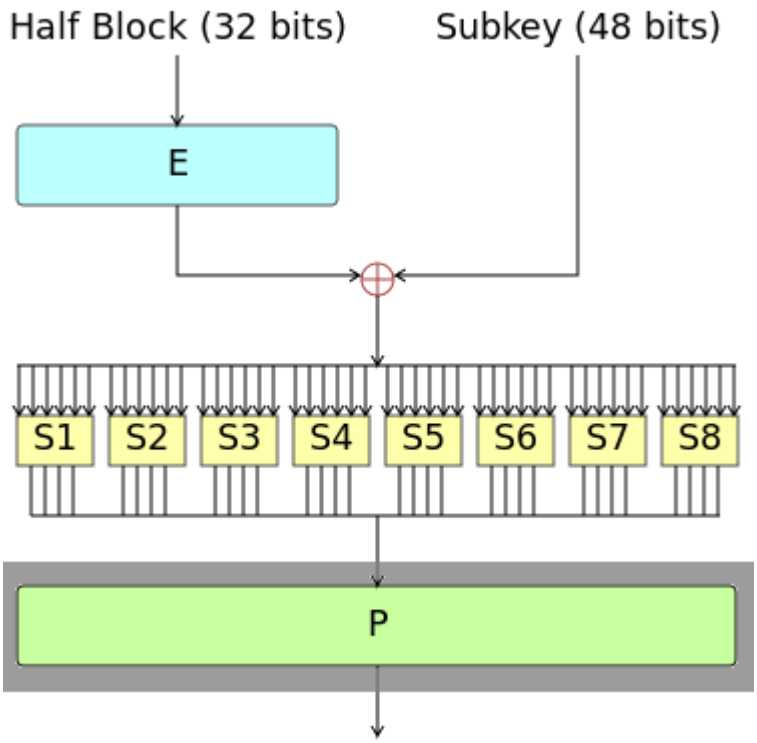
Pozor! Každý S-box má odlišnou tabulku, tabulky pro všechny S-boxy naleznete zde

Jak jsem již zmínil, tak výsledek každého S-boxu je čtyř-bitové číslo, ale na wikipedii jsou hodnoty výsledků jednotlivých S-boxů v desítkové (decimální) soustavě. Takže ještě musíte čísla převést do binární. Pozor na to, když například výsledkem bude v desítkové soustavě `2`, tak v binární bude `0010`. Výsledek musí mít vždy **čtyři bity!**

Výsledkem substituce je 32 bitů.

Permutace (P)

32bitový výsledek z předchozích osmi S-boxů se přehází v finálním P-boxu. To z toho důvodu, aby po permutaci byly bity z jednotlivých S-boxů rovnoměrně rozprostřeny pro vstup do S-boxů v dalším opakování funkce F.



Opět se řídí tabulkou a funguje na identickém principu jako IP nebo FP. Udává, ze kterého bitu vstupu se bere bit výstupu. První buňka (levá, horní) značí první bit výstupu a obsah buňky udává, z kolikátého bitu vstupu se berou data.

16	7	20	21	29	12	28	17
1	15	23	26	5	18	31	10
2	8	24	14	32	27	3	9
19	13	30	6	22	11	4	25

Tabulku čtěte po řádcích. První bit výstupu se bere z 16. bitu vstupu, druhý bit výstupu se bere z 7. bitu vstupu... až poslední 32. bit výstupu se bere z 25. bitu vstupu.

Dešifrování

Krása symetrického algoritmu, jako je DES je v tom, že pro dešifrování stačí v podstatě udělat pouze všechny kroky pozpátku.

To be continued

V příští úloze se podíváme na kryptografické útoky na DES, jako třeba Deep Crack, jak NSA i IBM znalo techniky kryptoanalýzy, které byly veřejností objeveny až o mnoho let později a na nástupce DESu, jako Triple DES a dodnes používané AES.