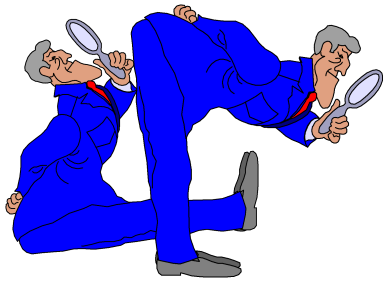
7. Šifrový standard DES a jeho kolegové

Jan Paseka

Ústav matematiky a statistiky
Masarykova univerzita

29. listopadu 2021

O čem to bude



Úvod I

FI Obrovský nárůst počítačové komunikace a nástup elektronických bankovních systémů v sedmdesátých letech byly jedním z hlavních faktorů pro rozhodnutí exekutivy Spojených států amerických na zajištění standardu pro bezpečný a spolehlivý transfer dat Federální banky a ostatních bank.

Úvod I

FI Obrovský nárůst počítačové komunikace a nástup elektronických bankovních systémů v sedmdesátých letech byly jedním z hlavních faktorů pro rozhodnutí exekutivy Spojených států amerických na zajištění standardu pro bezpečný a spolehlivý transfer dat Federální banky a ostatních bank.

Bylo tedy potřeba zajistitit ochranu dat jak v počítačích zpracovávaných a tamtéž ukládaných, tak i dat počítači přenášených.

Úvod I

FI Obrovský nárůst počítačové komunikace a nástup elektronických bankovních systémů v sedmdesátých letech byly jedním z hlavních faktorů pro rozhodnutí exekutivy Spojených států amerických na zajištění standardu pro bezpečný a spolehlivý transfer dat Federální banky a ostatních bank.

Bylo tedy potřeba zajistitit ochranu dat jak v počítačích zpracovávaných a tamtéž ukládaných, tak i dat počítači přenášených.

Soutěž na tvorbu šifrovacího algoritmu vyhlásilo ministerstvo obchodu Spojených států amerických v roce 1973. Organizoval ji National Bureau of Standards (NBS).

Úvod I

FI Obrovský nárůst počítačové komunikace a nástup elektronických bankovních systémů v sedmdesátých letech byly jedním z hlavních faktorů pro rozhodnutí exekutivy Spojených států amerických na zajištění standardu pro bezpečný a spolehlivý transfer dat Federální banky a ostatních bank.

Bylo tedy potřeba zajistitit ochranu dat jak v počítačích zpracovávaných a tamtéž ukládaných, tak i dat počítači přenášených.

Soutěž na tvorbu šifrovacího algoritmu vyhlásilo ministerstvo obchodu Spojených států amerických v roce 1973. Organizoval ji National Bureau of Standards (NBS).

Požadovaným vlastnostem však nevyhověl žádný ze zaslaných algoritmů - základní idea byla, že šifrovací proces by mělo být možno provádět na malém čipu.

Úvod II

Důsledkem pak by byla masová výroba těchto čipů, jejichž použití by zajišťovalo naprostou bezpečnost transferu dat.

Úvod II

Důsledkem pak by byla masová výroba těchto čipů, jejichž použití by zajišťovalo naprostou bezpečnost transferu dat.

Přitom sám algoritmus měl být bezpečný, pochopitelný, dostupný, výkonný, jeho bezpečnost neměla záviset na utajení algoritmu, vhodný pro nejrůznější aplikace a ekonomický při elektronické realizaci – tj. čip by měl být levný. V srpnu 1974 byla výzva opakována.

Úvod II

Důsledkem pak by byla masová výroba těchto čipů, jejichž použití by zajišťovalo naprostou bezpečnost transferu dat.

Přitom sám algoritmus měl být bezpečný, pochopitelný, dostupný, výkonný, jeho bezpečnost neměla záviset na utajení algoritmu, vhodný pro nejrůznější aplikace a ekonomický při elektronické realizaci – tj. čip by měl být levný. V srpnu 1974 byla výzva opakována.

Tentokrát se algoritmus podařilo nalézt. Bylo akceptováno šifrovací schéma navržené firmou IBM. Její výzkumný tým pod vedením dr. Tuchmana jej založil na zdokonalení šifrovacího algoritmu LUCIFER, který IBM používala pro své potřeby.

Úvod II

Důsledkem pak by byla masová výroba těchto čipů, jejichž použití by zajišťovalo naprostou bezpečnost transferu dat.

Přitom sám algoritmus měl být bezpečný, pochopitelný, dostupný, výkonný, jeho bezpečnost neměla záviset na utajení algoritmu, vhodný pro nejrůznější aplikace a ekonomický při elektronické realizaci – tj. čip by měl být levný. V srpnu 1974 byla výzva opakována.

Tentokrát se algoritmus podařilo nalézt. Bylo akceptováno šifrovací schéma navržené firmou IBM. Její výzkumný tým pod vedením dr. Tuchmana jej založil na zdokonalení šifrovacího algoritmu LUCIFER, který IBM používala pro své potřeby.

Na rozdíl od algoritmu LUCIFER, jež používal klíč o délce 128 bitů, délka klíče navržená pro NBS byla 64 bitů a z toho bylo 8 bitů odstraněno hned na začátku šifrování.

Úvod III

NBS, IBM a NSA (National Security Agency) se dohodly, že NSA provede ohodnocení bezpečnosti DES (Data Encryption Standars), jak byl později nový algoritmus nazván, a IBM umožní jeho bezplatné využívání na území Spojených států amerických. DES byl patentován 24.2. 1975 a v březnu téhož roku byl zveřejněn pro všeobecné veřejné hodnocení.

Úvod III

NBS, IBM a NSA (National Security Agency) se dohodly, že NSA provede ohodnocení bezpečnosti DES (Data Encryption Standars), jak byl později nový algoritmus nazván, a IBM umožní jeho bezplatné využívání na území Spojených států amerických. DES byl patentován 24.2. 1975 a v březnu téhož roku byl zveřejněn pro všeobecné veřejné hodnocení.

Přes některé výhrady byl 23.11. 1976 přijat jako federální standard a jako takový zveřejněn 15.1. 1977. Algoritmus byl určen pro ochranu neutajovaných dat v civilním sektoru i vládních institucí vyjma ozbrojených složek, kde nemohl být používán ani k ochraně neutajovaných dat. Předpokládaná doba použití byla 10-15 let.

Úvod III

NBS, IBM a NSA (National Security Agency) se dohodly, že NSA provede ohodnocení bezpečnosti DES (Data Encryption Standars), jak byl později nový algoritmus nazván, a IBM umožní jeho bezplatné využívání na území Spojených států amerických. DES byl patentován 24.2. 1975 a v březnu téhož roku byl zveřejněn pro všeobecné veřejné hodnocení.

Přes některé výhrady byl 23.11. 1976 přijat jako federální standard a jako takový zveřejněn 15.1. 1977. Algoritmus byl určen pro ochranu neutajovaných dat v civilním sektoru i vládních institucí vyjma ozbrojených složek, kde nemohl být používán ani k ochraně neutajovaných dat. Předpokládaná doba použití byla 10-15 let.

Schválen jako šifrovací standard (***Federal Information Processing Standard - FIPS 46 – 3***) v USA.

Úvod IV

DES vyhovuje normě ANSI z roku 1980 (ANSI X3, 92 -1981, pod názvem ***Data Encryption Algorithm, American National Standards Institute, New York, 1980***). Proto je také někdy citován jako DEA.

Úvod IV

DES vyhovuje normě ANSI z roku 1980 (ANSI X3, 92 -1981, pod názvem ***Data Encryption Algorithm, American National Standards Institute, New York, 1980***). Proto je také někdy citován jako DEA.

Algoritmus měl být po svém přijetí v roce 1977 každých pět let hodnocen a jeho platnost potvrzována NBS.

Úvod IV

DES vyhovuje normě ANSI z roku 1980 (ANSI X3, 92 -1981, pod názvem ***Data Encryption Algorithm, American National Standards Institute, New York, 1980***). Proto je také někdy citován jako DEA.

Algoritmus měl být po svém přijetí v roce 1977 každých pět let hodnocen a jeho platnost potvrzována NBS.

V roce 1984 zahájila NSA program "Commercial COMSEC Endorsement Program" (CCEP), určený pro zabezpečování ochrany vládních informací, v rámci něhož mělo být připraveno i nahrazování DES.

Úvod IV

DES vyhovuje normě ANSI z roku 1980 (ANSI X3, 92 -1981, pod názvem ***Data Encryption Algorithm, American National Standards Institute, New York, 1980***). Proto je také někdy citován jako DEA.

Algoritmus měl být po svém přijetí v roce 1977 každých pět let hodnocen a jeho platnost potvrzována NBS.

V roce 1984 zahájila NSA program "Commercial COMSEC Endorsement Program" (CCEP), určený pro zabezpečování ochrany vládních informací, v rámci něhož mělo být připraveno i nahrazování DES.

Byly vyvinuty hardwarové bezpečnostní moduly, provádějící šifrové algoritmy navržené pro tentokrát NSA. Tyto algoritmy byly typu 1 a typu 2.

Úvod V

První byl určen pro utajované vládní informace a druhý pro neutajované vládní informace (měl nahradit DES). Později byla aplikace zařízení s algoritmem typu 2 rozšířena i na soukromý sektor.
Po 1.1. 1988 neměla už NSA v úmyslu doporučovat DES pro vládní použití. Bylo však zjištěno, že by to způsobilo značné potíže v bankovním sektoru. DES byl v této době už masově používán v nejrůznějších zařízeních včetně mezinárodního bankovního spojení. To vedlo ministerstvo obchodu USA ke zmírnění stanoviska NSA.

Úvod V

První byl určen pro utajované vládní informace a druhý pro neutajované vládní informace (měl nahradit DES). Později byla aplikace zařízení s algoritmem typu 2 rozšířena i na soukromý sektor.
Po 1.1. 1988 neměla už NSA v úmyslu doporučovat DES pro vládní použití. Bylo však zjištěno, že by to způsobilo značné potíže v bankovním sektoru. DES byl v této době už masově používán v nejrůznějších zařízeních včetně mezinárodního bankovního spojení. To vedlo ministerstvo obchodu USA ke zmírnění stanoviska NSA.

V lednu 1988 NBS znovu potvrdil možnost používat DES v dalších 5 letech, avšak ne pro federální nefinanční použití. Ve federálních nefinančních institucích musel být DES nahrazován algoritmy vyvinutými NSA v rámci CCEP.

Úvod VI

Algoritmy jsou utajovány a nesmí být exportovány ze Spojených států amerických. Čipy, které je realizují, mají ochranný kryt.

Úvod VI

Algoritmy jsou utajovány a nesmí být exportovány ze Spojených států amerických. Čipy, které je realizují, mají ochranný kryt.

Výrobci produktů musí při jejich výrobě dodržovat zvláštní postup stanovený NSA. Tato opatření na ochranu nových algoritmů přinášejí do celé koncepce jejich využívání velké rozpory.

Úvod VI

Algoritmy jsou utajovány a nesmí být exportovány ze Spojených států amerických. Čipy, které je realizují, mají ochranný kryt.

Výrobci produktů musí při jejich výrobě dodržovat zvláštní postup stanovený NSA. Tato opatření na ochranu nových algoritmů přinášejí do celé koncepce jejich využívání velké rozpory.

Jejich softwarová implementace by nebyla schválena, protože by zde nemohla být zajištěna ochrana algoritmu před jeho odhalením. Přitom v mnoha aplikacích je softwarová implementace nezbytná.

Úvod VI

Algoritmy jsou utajovány a nesmí být exportovány ze Spojených států amerických. Čipy, které je realizují, mají ochranný kryt.

Výrobci produktů musí při jejich výrobě dodržovat zvláštní postup stanovený NSA. Tato opatření na ochranu nových algoritmů přinášejí do celé koncepce jejich využívání velké rozpory.

Jejich softwarová implementace by nebyla schválena, protože by zde nemohla být zajištěna ochrana algoritmu před jeho odhalením. Přitom v mnoha aplikacích je softwarová implementace nezbytná.

Otázkou zůstávají i mezinárodní spoje, kde se předpokládá vývoz těchto zařízení. Je pravděpodobné, že dnes je už povolen přísně regulovaný vývoz, i když algoritmy zůstávají nadále utajeny.

Úvod VII

Vzhledem k tomu, že uvedená obranná opatření fungují velmi dobře, o zařízeních nevíme téměř nic, dokonce neznáme, ani jejich výkonové charakteristiky.

Úvod VII

Vzhledem k tomu, že uvedená obranná opatření fungují velmi dobře, o zařízeních nevíme téměř nic, dokonce neznáme, ani jejich výkonové charakteristiky.

Také není dosud nic známo o rozhodnutí, které mělo být přijato NBS v únoru r. 1993, týkajícím se dalšího existence DES ve finančním sektoru pro léta 1993–1997.

Úvod VII

Vzhledem k tomu, že uvedená obranná opatření fungují velmi dobře, o zařízeních nevíme téměř nic, dokonce neznáme, ani jejich výkonové charakteristiky.

Také není dosud nic známo o rozhodnutí, které mělo být přijato NBS v únoru r. 1993, týkajícím se dalšího existence DES ve finančním sektoru pro léta 1993–1997.

DES se od roku 1977 stal ***nejrozšířenějším kryptografickým systémem*** ve světě. Je všeobecným nástrojem bezpečnosti ve veřejném i soukromém sektoru.

Úvod VII

Vzhledem k tomu, že uvedená obranná opatření fungují velmi dobře, o zařízeních nevíme téměř nic, dokonce neznáme, ani jejich výkonové charakteristiky.

Také není dosud nic známo o rozhodnutí, které mělo být přijato NBS v únoru r. 1993, týkajícím se dalšího existence DES ve finančním sektoru pro léta 1993–1997.

DES se od roku 1977 stal ***nejrozšířenějším kryptografickým systémem*** ve světě. Je všeobecným nástrojem bezpečnosti ve veřejném i soukromém sektoru.

Ve finančním sektoru se používá k ochraně všech aspektů síťové komunikace a autentizace a de facto se stal mezinárodním standardem. Je včleněn do maloobchodních i velkoobchodních systémů, je přístupný v nedrahém hardwaru a jako volný software.

Úvod VIII

Pouze ve Spojených státech amerických vývoz hardwaru i softwaru s DES dosud podléhá kontrole. DES je dostupný ve všech možných formách: ve formě čipů, zásuvkových modulů do PC nebo celých šifrovacích jednotek.

Úvod VIII

Pouze ve Spojených státech amerických vývoz hardwaru i softwaru s DES dosud podléhá kontrole. DES je dostupný ve všech možných formách: ve formě čipů, zásuvkových modulů do PC nebo celých šifrovacích jednotek.

Jsou k dispozici oficiální programy pro softwarovou realizaci a pro kontrolu správnosti jeho realizace v zařízeních. Standardizaci na bázi DES podporuje pět největších standardizačních organizací: ABA, ANSI, GSA, ISO a NBS.

Úvod VIII

Pouze ve Spojených státech amerických vývoz hardwaru i softwaru s DES dosud podléhá kontrole. DES je dostupný ve všech možných formách: ve formě čipů, zásuvkových modulů do PC nebo celých šifrovacích jednotek.

Jsou k dispozici oficiální programy pro softwarovou realizaci a pro kontrolu správnosti jeho realizace v zařízeních. Standardizaci na bázi DES podporuje pět největších standardizačních organizací: ABA, ANSI, GSA, ISO a NBS.

Standardy NBS jsou používány jako základ standardů dalších organizací. DES se používá pro šifrování PIN (Personal Identification Number) v různých druzích platebních, přístupových aj. karet.

Úvod IX

Také v několika amerických vládních organizacích vytváří DES základní mechanismus ochrany (ministerstvo ekonomiky, ministerstvo spravedlnosti). V USA na bázi DES vznikl kryptografický průmysl.

Úvod IX

Také v několika amerických vládních organizacích vytváří DES základní mechanismus ochrany (ministerstvo ekonomiky, ministerstvo spravedlnosti). V USA na bázi DES vznikl kryptografický průmysl.

Na konci roku 1999 bylo doporučeno, aby se přešlo z DES na její vylepšenou verzi 3DES (triple DES; trojitý DES). 3DES je trojnásobná aplikace DES algoritmu, pokaždé s jiným klíčem. Klíč šifry 3DES je tedy 168 bitů dlouhý.

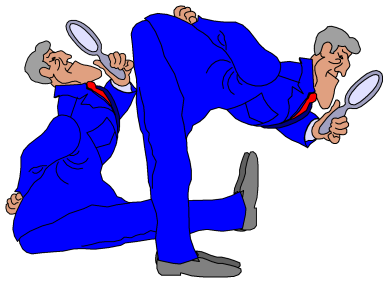
Úvod IX

Také v několika amerických vládních organizacích vytváří DES základní mechanismus ochrany (ministerstvo ekonomiky, ministerstvo spravedlnosti). V USA na bázi DES vznikl kryptografický průmysl.

Na konci roku 1999 bylo doporučeno, aby se přešlo z DES na její vylepšenou verzi 3DES (triple DES; trojitý DES). 3DES je trojnásobná aplikace DES algoritmu, pokaždé s jiným klíčem. Klíč šifry 3DES je tedy 168 bitů dlouhý.

I když je algoritmus DES již vlastně minulostí, neboť jej lze snadno prolomit, 3DES je stále nasazován v mnoha aplikacích. Například Secure shell (ssh), shadow hesla v Unixu, šifrování uživatelských hesel v databázových serverech Sybase, Nortel VPN, apod.

O čem to bude



systémech

Bloková šifra I

DES patří do třídy blokových šifer. Otevřený text (OT) je rozdělen na bloky po 64 bitech.

Bloková šifra I

DES patří do třídy blokových šifer. Otevřený text (OT) je rozdělen na bloky po 64 bitech.

Každý z těchto bloků M otevřeného textu je jako celek zpracováván na blok C šifrového textu (ŠT) také o délce 64 bitů.

Bloková šifra I

DES patří do třídy blokových šifer. Otevřený text (OT) je rozdělen na bloky po 64 bitech.

Každý z těchto bloků M otevřeného textu je jako celek zpracováván na blok C šifrového textu (ŠT) také o délce 64 bitů.

Píšeme jako obvykle

$$C = E_K(M), \quad M = D_K(C),$$

kde K je klíč o délce 56 bitů (vznikne z osmibajtového klíčového slova vynecháním jeho paritních bitů).

Bloková šifra I

DES patří do třídy blokových šifer. Otevřený text (OT) je rozdělen na bloky po 64 bitech.

Každý z těchto bloků M otevřeného textu je jako celek zpracováván na blok C šifrového textu (ŠT) také o délce 64 bitů.

Píšeme jako obvykle

$$C = E_K(M), \quad M = D_K(C),$$

kde K je klíč o délce 56 bitů (vznikne z osmibajtového klíčového slova vynecháním jeho paritních bitů).

Zpracování bloku M probíhá v 16 krocích. V každém z nich je z klíče K vybráno podle určitého postupu 48 bitů tvořících pracovní klíč K_i.

Bloková šifra I

Vlastní blokový algoritmus při šifrování zpracovává blok M tak, že M je nejprve podroben ***počáteční permutací*** IP, která permutuje všech 64 bitů, a poté je rozdělen na pravou polovinu R_0 a levou polovinu L_0, každá o 32 bitech.

Bloková šifra I

Vlastní blokový algoritmus při šifrování zpracovává blok M tak, že M je nejprve podroben ***počáteční permutací*** IP, která permutuje všech 64 bitů, a poté je rozdělen na pravou polovinu R_0 a levou polovinu L_0, každá o 32 bitech.

Pak probíhá 16 totožných kroků, kdy je z dvojice (L_i, R_i) za účasti pracovního klíče K_i vytvořena dvojice (L_{i+1}, R_{i+1}), $i = 0, 1, \ldots, 15$.

Bloková šifra I

Vlastní blokový algoritmus při šifrování zpracovává blok M tak, že M je nejprve podroben ***počáteční permutací*** IP, která permutuje všech 64 bitů, a poté je rozdělen na pravou polovinu R_0 a levou polovinu L_0, každá o 32 bitech.

Pak probíhá 16 totožných kroků, kdy je z dvojice (L_i, R_i) za účasti pracovního klíče K_i vytvořena dvojice (L_{i+1}, R_{i+1}), $i = 0, 1, \ldots, 15$.

R_i je nejprve rozšířen z 32 bitů na 48 bitů prostřednictvím expanze E tak, že

$$E(R_i) = E(r_1, r_2, \ldots, r_{31}, r_{32}) = (r_{32}, r_1, r_2, r_3, r_4, r_5, r_4, r_5, r_6, r_7, r_8, r_9, \ldots, r_{28}, r_{29}, r_{30}, r_{31}, r_{32}, r_1), \quad (2.1)$$

a k výsledku je v modulu 2 tj. bit po bitu přečten klíč K_i.

Bloková šifra II

Výstup $E(R_i) \oplus K_i$ je rozdělen do osmi částí po 6 bitech, které procházejí přes substituční boxy S_1 až S_8.

Bloková šifra II

Výstup $E(R_i) \oplus K_i$ je rozdělen do osmi částí po 6 bitech, které procházejí přes substituční boxy S_1 až S_8.

Tyto ***S-boxy*** jsou v podstatě pamětí ROM 6×4 bity a výstup je tedy celkem $8 \times 4 = 32$-bitový.

Bloková šifra II

Výstup $E(R_i) \oplus K_i$ je rozdělen do osmi částí po 6 bitech, které procházejí přes substituční boxy S_1 až S_8.

Tyto ***S-boxy*** jsou v podstatě pamětí ROM 6×4 bity a výstup je tedy celkem $8 \times 4 = 32$-bitový.

Většinou se tyto boxy popisují i zadávají tak, že krajní bity vstupu (1. a 6. bit) vybírají jeden ze čtyř možných boxů 4×4 bity a výstup těchto boxů se uvádí v tabulce.

Bloková šifra II

Výstup $E(R_i) \oplus K_i$ je rozdělen do osmi částí po 6 bitech, které procházejí přes substituční boxy S_1 až S_8.

Tyto ***S-boxy*** jsou v podstatě pamětí ROM 6×4 bity a výstup je tedy celkem 8×4=32-bitový.

Většinou se tyto boxy popisují i zadávají tak, že krajní bity vstupu (1. a 6. bit) vybírají jeden ze čtyř možných boxů 4×4 bity a výstup těchto boxů se uvádí v tabulce.

Přitom se jejich vstupní i výstupní 4-bitová hodnota pro jednoduchost zapisuje dekadicky jako 0 až 15. Jsou to permutace na množině $\{0, 1, \ldots, 15\}$.

Bloková šifra II

Výstup $E(R_i) \oplus K_i$ je rozdělen do osmi částí po 6 bitech, které procházejí přes substituční boxy S_1 až S_8.

Tyto ***S-boxy*** jsou v podstatě pamětí ROM 6×4 bity a výstup je tedy celkem $8 \times 4 = 32$-bitový.

Většinou se tyto boxy popisují i zadávají tak, že krajní bity vstupu (1. a 6. bit) vybírají jeden ze čtyř možných boxů 4×4 bity a výstup těchto boxů se uvádí v tabulce.

Přitom se jejich vstupní i výstupní 4-bitová hodnota pro jednoduchost zapisuje dekadicky jako 0 až 15. Jsou to permutace na množině $\{0, 1, \ldots, 15\}$.

Výstup z S-boxů, tj. $8 \times 4 = 32$ bity, je upraven ***permutací*** P (32 na 32 bitů).

Bloková šifra III

Vzniklé 32-bitové slovo je označováno $f(R_i, K_i)$, neboť je funkcí klíče K_i a slova R_i.

Bloková šifra III

Vzniklé 32-bitové slovo je označováno $f(R_i, K_i)$, neboť je funkcí klíče K_i a slova R_i.

Poslední operace v i-tém kroku je

$$L_{i+1} = R_i \qquad R_{i+1} = L_i \oplus f(R_i, K_i). \tag{2.2}$$

Po proběhnutí 16. kroku je provedena záměna L_{16} a R_{16} a 64-bitový blok (R_{16}, L_{16}) je permutován ***permutací*** IP^{-1} (permutací inverzní k IP).

Bloková šifra III

Vzniklé 32-bitové slovo je označováno $f(R_i, K_i)$, neboť je funkcí klíče K_i a slova R_i.

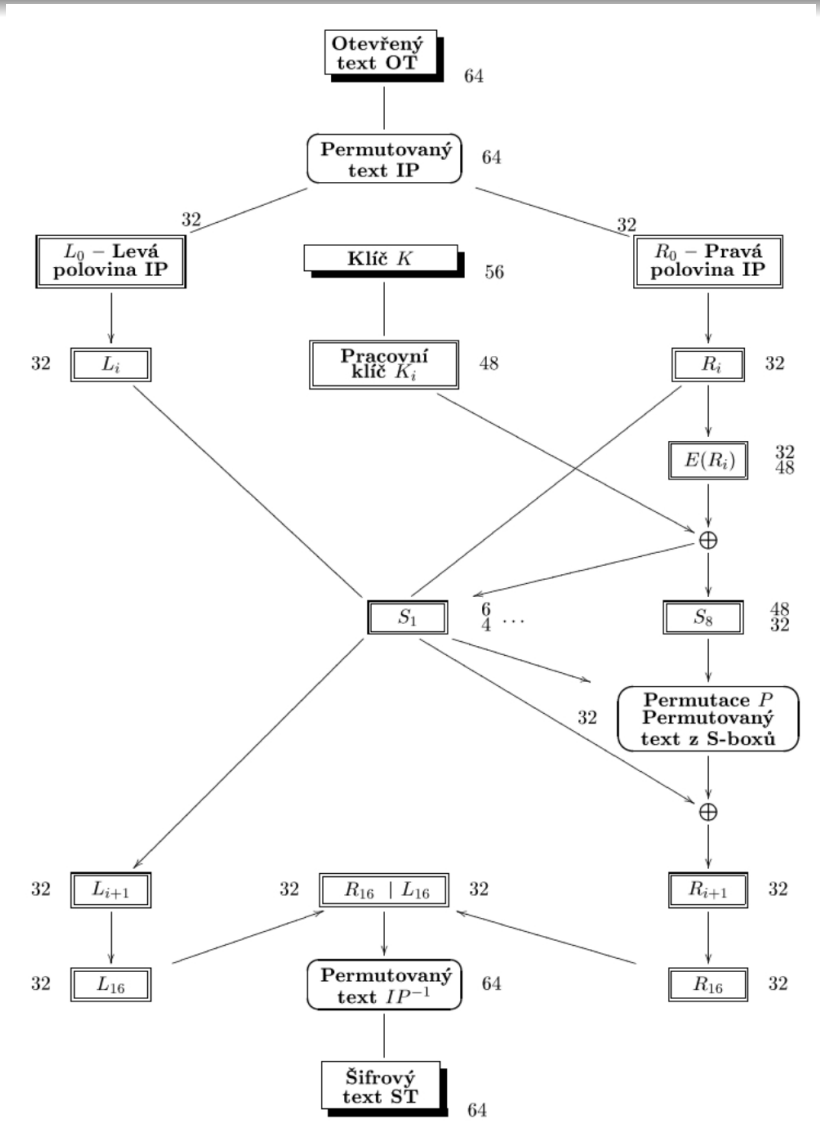
Poslední operace v i-tém kroku je

$$L_{i+1} = R_i \qquad R_{i+1} = L_i \oplus f(R_i, K_i). \tag{2.2}$$

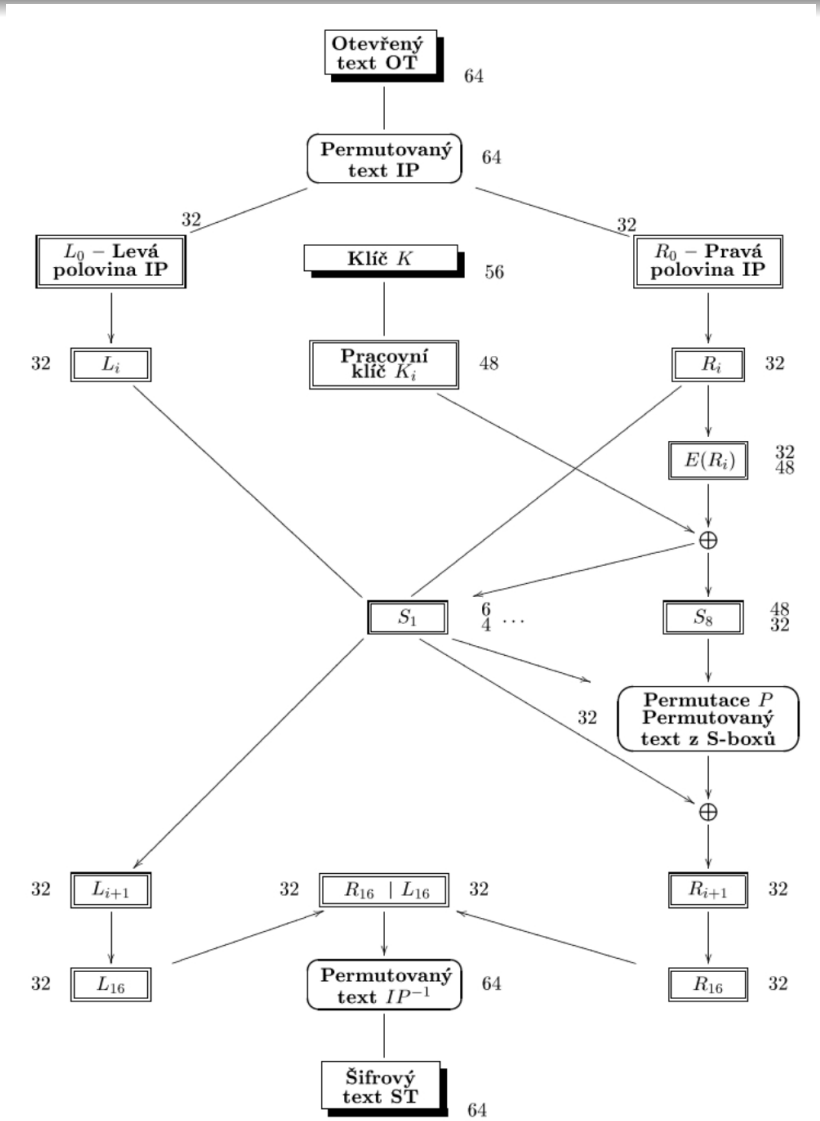
Po proběhnutí 16. kroku je provedena záměna L_{16} a R_{16} a 64-bitový blok (R_{16}, L_{16}) je permutován ***permutací*** IP^{-1} (permutací inverzní k IP).

Výsledek je již $C = E_K(M)$.

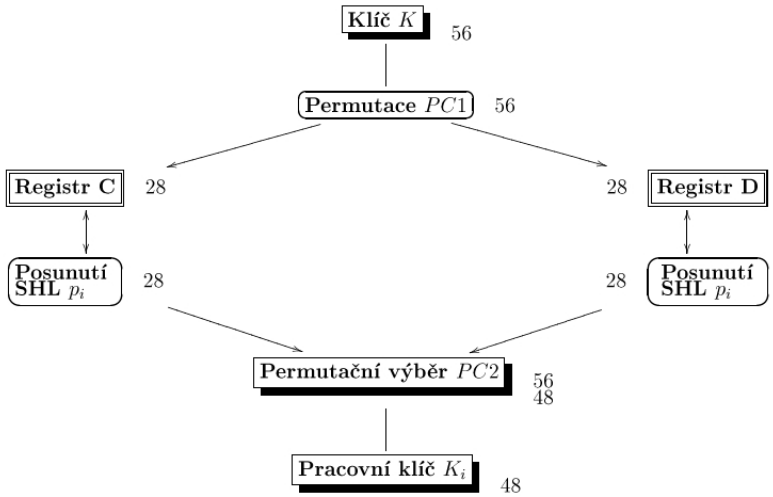
Bloková šifra IV



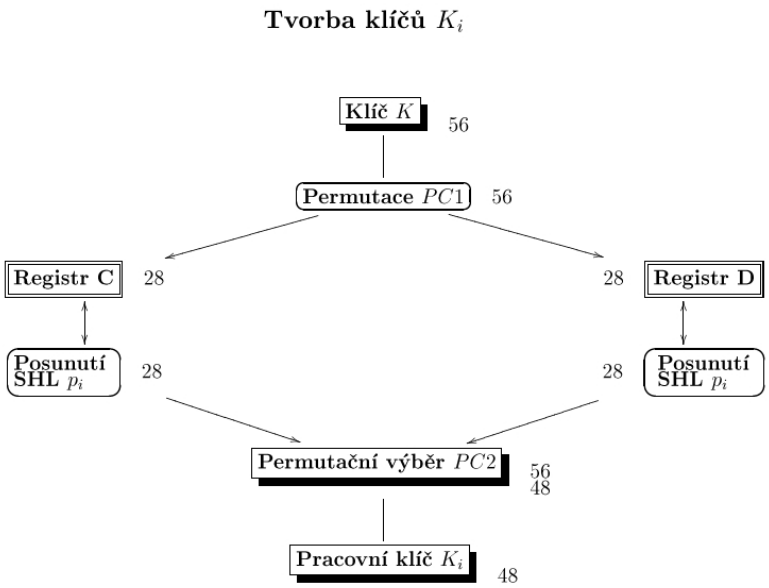
Bloková šifra IV



Tvorba klíčů K_i



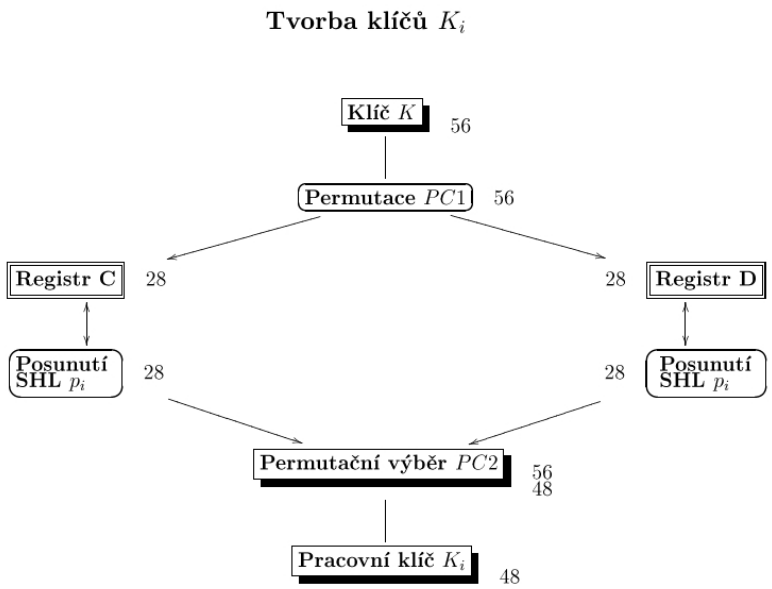
Bloková šifra V



Tvorba pracovních klíčů K_i probíhá při šifrování tak, že klíč K o 56 bitech je podroben permutaci $PC1$ a poté je naplněn do dvou 28-bitových ***registrů*** C a D.

Obsah registrů C, D je v každém kroku i, $i = 0, 1, \ldots, 15$ cyklicky posunut doleva o p_i bitů.

Bloková šifra V

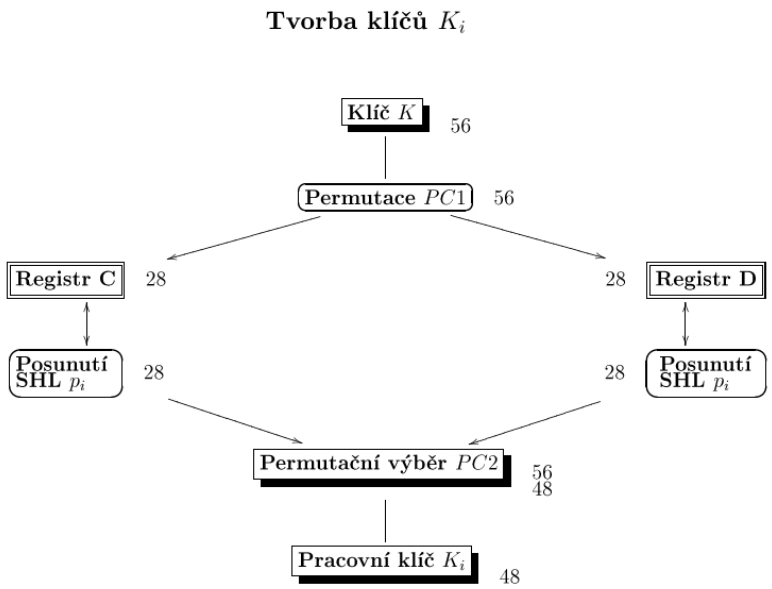


Tvorba pracovních klíčů K_i probíhá při šifrování tak, že klíč K o 56 bitech je podroben permutaci $PC1$ a poté je naplněn do dvou 28-bitových ***registrů*** C a D.

Obsah registrů C, D je v každém kroku i, $i = 0, 1, \ldots, 15$ cyklicky posunut doleva o p_i bitů.

Posun p_i je v krocích 0, 1, 8 a 15 jednobitový a ve zbývajících dvoubitový.

Bloková šifra V



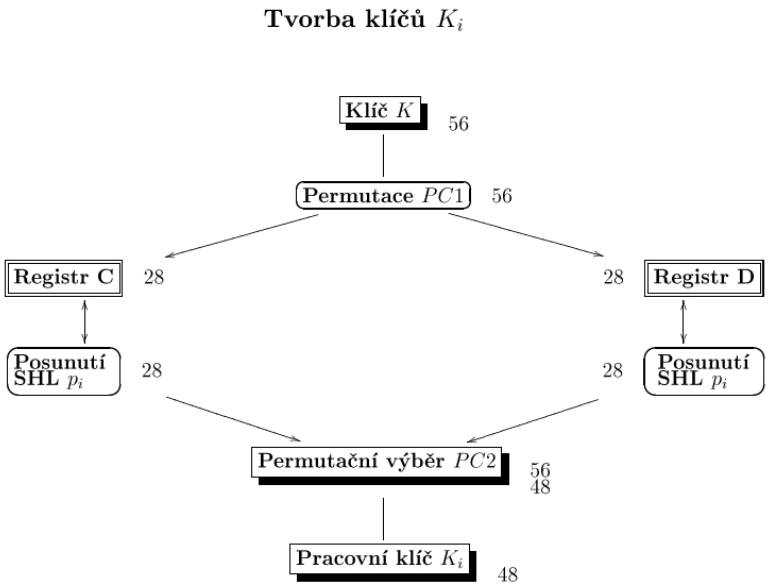
Tvorba pracovních klíčů K_i probíhá při šifrování tak, že klíč K o 56 bitech je podroben permutaci $PC1$ a poté je naplněn do dvou 28-bitových ***registrů*** C a D.

Obsah registrů C, D je v každém kroku i, $i = 0, 1, \ldots, 15$ cyklicky posunut doleva o p_i bitů.

Posun p_i je v krocích 0, 1, 8 a 15 jednobitový a ve zbývajících dvoubitový.

Výsledné 56-bitové zřetězení registrů C, D je podrobeno ***permutačnímu výběru*** $PC2$.

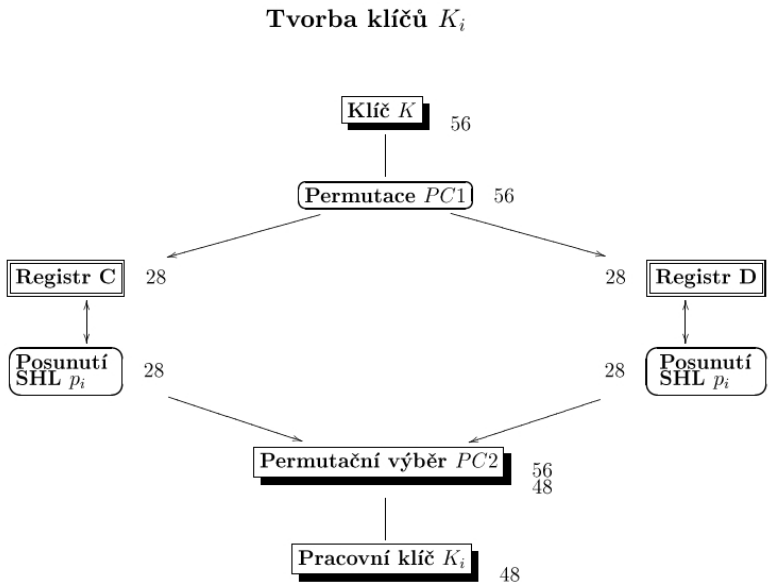
Bloková šifra VI



PC2 svůj vstup 56 bitů nejen permutuje, ale i redukuje na 48 bitů vynecháním některých bitů.

Výstup z *PC2* už tvoří pracovní klíč K_i. Klíče K_i lze vytvářet buď souběžně se zpracováním otevřeného textu nebo předem.

Bloková šifra VI

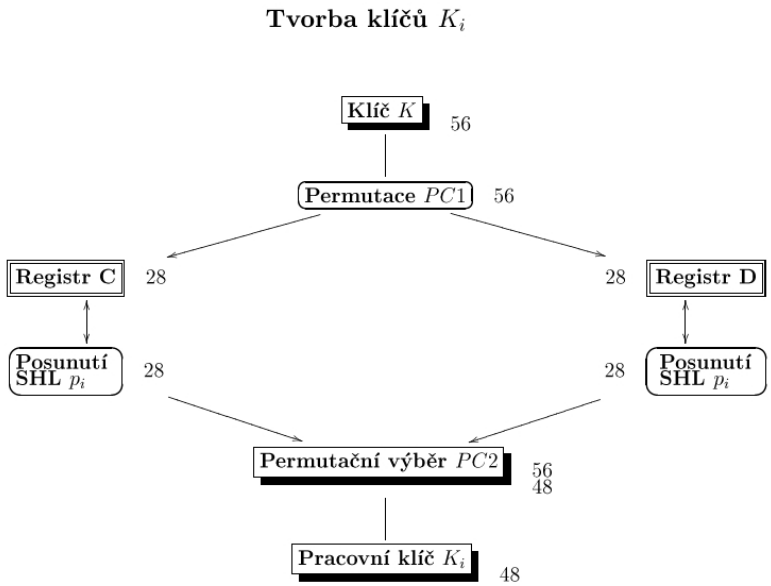


PC2 svůj vstup 56 bitů nejen permutuje, ale i redukuje na 48 bitů vynecháním některých bitů.

Výstup z *PC2* už tvoří pracovní klíč K_i. Klíče K_i lze vytvářet buď souběžně se zpracováním otevřeného textu nebo předem.

Postup formování klíčů K_i je takový, že v uvedených ***16×48=768*** bitech je každý bit klíče K obsažen ***12 až 15-krát*** a objevuje se na různých pozicích.

Bloková šifra VI



PC2 svůj vstup 56 bitů nejen permutuje, ale i redukuje na 48 bitů vynecháním některých bitů.

Výstup z *PC2* už tvoří pracovní klíč K_i. Klíče K_i lze vytvářet buď souběžně se zpracováním otevřeného textu nebo předem.

Postup formování klíčů K_i je takový, že v uvedených ***16×48=768*** bitech je každý bit klíče K obsažen ***12 až 15-krát*** a objevuje se na různých pozicích.

Tímto je popis šifrovací části blokového algoritmu uzavřen.

Bloková šifra VII

Filozofie algoritmu je taková, aby při dešifrování nemuselo být použito zcela jiné hardwarové schéma.

Bloková šifra VII

Filozofie algoritmu je taková, aby při dešifrování nemuselo být použito zcela jiné hardwarové schéma.

Dešifrování $M = D_K(C)$ ***probíhá stejným postupem jako šifrování!***

Bloková šifra VII

Filozofie algoritmu je taková, aby při dešifrování nemuselo být použito zcela jiné hardwarové schéma.

Dešifrování $M = D_K(C)$ ***probíhá stejným postupem jako šifrování!***

Pouze pořadí výběru klíčů K_i je obrácené - místo $K_0, K_1, \ldots, K_{15}$ se používá pořadí $K_{15}, K_{14}, \ldots, K_2, K_1, K_0$.

Bloková šifra VII

Filozofie algoritmu je taková, aby při dešifrování nemuselo být použito zcela jiné hardwarové schéma.

Dešifrování $M = D_K(C)$ ***probíhá stejným postupem jako šifrování!***

Pouze pořadí výběru klíčů K_i je obrácené - místo $K_0, K_1, \ldots, K_{15}$ se používá pořadí $K_{15}, K_{14}, \ldots, K_2, K_1, K_0$.

Vytváříme-li klíče postupně, pak ve výše uvedeném popisu se místo SHL musí použít SHR a tabulka posunů $(0, 1, 2, 2, 2, 2, 2, 2, 1, 2, 2, 2, 2, 2, 2, 1)$, jinak zůstane vše zachováno.

Bloková šifra VII

Filozofie algoritmu je taková, aby při dešifrování nemuselo být použito zcela jiné hardwarové schéma.

Dešifrování $M = D_K(C)$ ***probíhá stejným postupem jako šifrování!***

Pouze pořadí výběru klíčů K_i je obrácené - místo $K_0, K_1, \ldots, K_{15}$ se používá pořadí $K_{15}, K_{14}, \ldots, K_2, K_1, K_0$.

Vytváříme-li klíče postupně, pak ve výše uvedeném popisu se místo SHL musí použít SHR a tabulka posunů $(0, 1, 2, 2, 2, 2, 2, 2, 1, 2, 2, 2, 2, 2, 2, 1)$, jinak zůstane vše zachováno.
Tento princip zpracování (L_i, R_i) na (L_{i+1}, R_{i+1}) se nazývá ***Feistelův princip*** podle dr. Horsta Feistela, kryptologa IBM a vynálezce šifrovacího algoritmu LUCIFER.

Bloková šifra VIII - Tabulka permutací

	IP	PC1	PC2	P
1	58	57	14	16
2	50	49	17	7
3	42	41	11	20
4	34	33	24	21
5	26	25	1	29
6	18	17	5	12
7	10	9	3	28
8	2	1	28	17
9	60	58	15	1
10	52	50	6	15
11	44	42	21	23
12	36	34	10	26
13	28	26	23	5
14	20	18	19	18
15	12	10	12	31
16	4	2	4	10

	IP	PC1	PC2	P
17	62	59	26	2
18	54	51	8	8
19	46	43	16	24
20	38	35	7	14
21	30	27	27	32
22	22	19	20	27
23	14	11	13	3
24	6	3	2	9
25	64	69	41	19
26	56	52	52	13
27	48	44	31	30
28	40	36	37	6
29	32	63	47	22
30	24	55	55	11
31	16	47	30	4
32	8	39	40	25

Bloková šifra IX - Tabulka permutací

	IP	PC1	PC2
33	57	31	51
34	49	23	45
35	41	15	33
36	33	7	48
37	25	62	44
38	17	54	49
39	9	46	39
40	1	38	56
41	59	30	34
42	51	22	53
43	43	14	46
44	35	6	42
45	27	61	50
46	19	53	36
47	11	45	29
48	3	37	32

	IP	PC1	PC2
49	61	29	
50	53	21	
51	45	13	
52	37	5	
53	29	28	
54	21	20	
55	13	12	
56	5	4	
57	63		
58	55		
59	47		
60	39		
61	31		
62	23		
63	15		
64	7		

Bloková šifra X - Tabulka permutací

S-box S1

		(x_2, x_3, x_4, x_5)														
x_1	x_6	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**
0	**0**	14	4	13	1	2	15	11	8	3	10	6	12	5	9	0	7
0	**1**	0	15	7	4	14	2	13	1	10	6	12	11	9	5	3	8
1	**0**	4	1	14	8	13	6	2	11	15	12	9	7	3	10	5	0
1	**1**	15	12	8	2	4	9	1	7	5	11	3	14	10	0	6	13

S-box S2

		(x_2, x_3, x_4, x_5)														
x_1	x_6	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**
0	**0**	15	1	8	14	6	11	3	4	9	7	2	13	12	0	5	10
0	**1**	3	13	4	7	15	2	8	14	12	0	1	10	6	9	11	5
1	**0**	0	14	7	11	10	4	13	1	5	8	12	6	9	3	2	15
1	**1**	13	8	10	1	3	15	4	2	11	6	7	12	0	5	14	9

Bloková šifra X - Tabulka permutací

S-box S3

		(x_2, x_3, x_4, x_5)															
x_1	x_6	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**
0	**0**	10	0	9	14	6	3	15	5	1	13	12	7	11	4	2	8
0	**1**	13	7	0	9	3	4	6	10	2	8	5	14	12	11	15	1
1	**0**	13	6	4	9	8	15	3	0	11	1	2	12	5	10	14	7
1	**1**	1	10	13	0	6	9	8	7	4	15	14	3	11	5	2	12

S-box S4

		(x_2, x_3, x_4, x_5)															
x_1	x_6	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**
0	**0**	7	13	14	3	0	6	9	10	1	2	8	5	11	12	4	15
0	**1**	13	8	11	5	6	15	0	3	4	7	2	12	1	10	14	9
1	**0**	10	6	9	0	12	11	7	13	15	1	3	14	5	2	8	4
1	**1**	3	15	0	6	10	1	13	8	9	4	5	11	12	7	2	14

Bloková šifra XI - Tabulka permutací

S-box S5

		(x_2, x_3, x_4, x_5)															
x_1	x_6	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**
0	**0**	2	12	4	1	7	10	11	6	8	5	3	15	13	0	14	9
0	**1**	14	11	2	12	4	7	13	1	5	0	15	10	3	9	8	6
1	**0**	4	2	1	11	10	13	7	8	15	9	12	5	6	3	0	14
1	**1**	11	8	12	7	1	14	2	13	6	15	0	9	10	4	5	3

S-box S6

		(x_2, x_3, x_4, x_5)															
x_1	x_6	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**
0	**0**	12	1	10	15	9	2	6	8	0	13	3	4	14	7	5	11
0	**1**	10	15	4	2	7	12	9	5	6	1	13	14	0	11	3	8
1	**0**	9	14	15	5	2	8	12	3	7	0	4	10	1	13	11	6
1	**1**	4	3	2	12	9	5	15	10	11	14	1	7	6	0	8	13

Bloková šifra XI - Tabulka permutací

S-box S7

		(x_2, x_3, x_4, x_5)															
x_1	x_6	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**
0	**0**	4	11	2	14	15	0	8	13	3	12	9	7	5	10	6	1
0	**1**	13	0	11	7	4	9	1	10	14	3	5	12	2	15	8	6
1	**0**	1	11	4	13	12	3	7	14	10	15	6	8	0	5	9	2
1	**1**	6	11	13	8	1	4	10	7	9	5	0	15	14	2	3	12

S-box S8

		(x_2, x_3, x_4, x_5)															
x_1	x_6	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**
0	**0**	13	2	8	4	6	15	11	1	10	9	3	14	5	0	12	7
0	**1**	1	15	13	8	10	3	7	4	12	5	6	11	0	14	9	2
1	**0**	7	11	4	1	9	12	14	2	0	6	10	13	15	3	5	8
1	**1**	2	1	14	7	4	10	8	13	15	12	9	0	3	5	6	11

Použití NP-těžkých problémů I

V současné době neexistuje rychlý (tj. pracující v polynomiální době) algoritmus pro žádný NP-těžký problém, a pokud NP$\neq$P, takový algoritmus neexistuje. Je tedy přirozený nápad založit kryptosystém na NP-těžkém problému, aby rozbití šifrovacího systému bylo ekvivalentní nalezení rychlého algoritmu, který by řešil NP-těžký problém.

Použití NP-těžkých problémů I

V současné době neexistuje rychlý (tj. pracující v polynomiální době) algoritmus pro žádný NP-těžký problém, a pokud NP$\neq$P, takový algoritmus neexistuje. Je tedy přirozený nápad založit kryptosystém na NP-těžkém problému, aby rozbití šifrovacího systému bylo ekvivalentní nalezení rychlého algoritmu, který by řešil NP-těžký problém.

To je mj. základní idea pro DES. Uvažme následující výpočetní problém.

Algebraické rovnice nad $\mathbf{Z}_2$

Vstup: Polynomy $p_1, \ldots, p_k$ v proměnných $x_1, \ldots, x_n$ nad $\mathbf{Z}_2$.

Otázka: Mají polynomy společný nulový bod $(x_1, \ldots . x_n)$ v $\mathbf{Z}_2^n$?

Použití NP-těžkých problémů II

Například následující tři rovnice

$$\begin{aligned} x_1x_4x_6 + x_2x_4x_5 - 1 &= 0 \\ x_1x_2 + x_2x_3 + x_3x_4 - 1 &= 0 \\ x_1x_3 + x_4x_5 + x_1x_6 - 1 &= 0 \end{aligned}$$

mají společné řešení $(1, 0, 1, 1, 1, 1)$. Ačkoliv výše uvedený problém je lehký pro malá k a n, se zvětšováním těchto parametrů se obtížnost vyřešení neustále zvyšuje.

Použití NP-těžkých problémů II

Například následující tři rovnice

$$\begin{aligned} x_1x_4x_6 + x_2x_4x_5 - 1 &= 0 \\ x_1x_2 + x_2x_3 + x_3x_4 - 1 &= 0 \\ x_1x_3 + x_4x_5 + x_1x_6 - 1 &= 0 \end{aligned}$$

mají společné řešení $(1, 0, 1, 1, 1, 1)$. Ačkoliv výše uvedený problém je lehký pro malá k a n, se zvětšováním těchto parametrů se obtížnost vyřešení neustále zvyšuje.

Lze navíc dokázat, že platí

Věta 2.1

Problém rozhodnout, zda algebraický systém rovnic modulo 2 má řešení, je NP-těžký.

Použití NP-těžkých problémů II

Výše uvedená věta byla použita při tvorbě šifrovacího systému LUCIFER navrženého IBM a popsaného H. Feistelem v roce 1973.

Použití NP-těžkých problémů II

Výše uvedená věta byla použita při tvorbě šifrovacího systému LUCIFER navrženého IBM a popsaného H. Feistelem v roce 1973.

Systém pracuje následovně. Bez újmy na obecnosti lze předpokládat, že délka zprávy M je $2n$ (pokud by byla větší, použijeme rozdělení do bloků délky $2n$ a na každý blok aplikujeme šifrovací proceduru).

Použití NP-těžkých problémů II

Výše uvedená věta byla použita při tvorbě šifrovacího systému LUCIFER navrženého IBM a popsaného H. Feistelem v roce 1973.

Systém pracuje následovně. Bez újmy na obecnosti lze předpokládat, že délka zprávy M je $2n$ (pokud by byla větší, použijeme rozdělení do bloků délky $2n$ a na každý blok aplikujeme šifrovací proceduru).

Zprávu M rozdělíme na dva stejné bloky délky n, tj. $M = (M_0, M_1)$. Klíč K bude vektor $\mathbf{k}$, který určuje podklíče $\mathbf{k_1}, \mathbf{k_2}, \ldots, \mathbf{k_{d-1}}$. Pro $2 \leq i \leq d$, rekurzivně definujeme

$$M_i = M_{i-2} + f(\mathbf{k_{i-1}}, M_{i-1}), \tag{2.3}$$

kde f je nějaká libovolná pevně zvolená nelineární transformace a počítáme modulo 2.

Použití NP-těžkých problémů III

Závěrečný kryptogram C je určen vztahem

$$C = e(M, K) = (M_{d-1}, M_d). \tag{2.4}$$

Použití NP-těžkých problémů III

Závěrečný kryptogram C je určen vztahem

$$C = e(M, K) = (M_{d-1}, M_d). \tag{2.4}$$

Zprávu M lze získat z kryptogramu C opačným postupem – všechno, co musí příjemce zprávy provést, je obrácení pořadí 2.3.

Použití NP-těžkých problémů III

Závěrečný kryptogram C je určen vztahem

$$C = e(M, K) = (M_{d-1}, M_d). \tag{2.4}$$

Zprávu M lze získat z kryptogramu C opačným postupem – všechno, co musí příjemce zprávy provést, je obrácení pořadí 2.3.

Tedy

$$M_{i-2} = M_i + f(\mathbf{k_{i-1}}, M_{i-1}), \tag{2.5}$$

$d \geq i \geq 2$, až dospějeme k původní zprávě $M = (M_0, M_1)$.

Použití NP-těžkých problémů III

Závěrečný kryptogram C je určen vztahem

$$C = e(M, K) = (M_{d-1}, M_d). \tag{2.4}$$

Zprávu M lze získat z kryptogramu C opačným postupem – všechno, co musí příjemce zprávy provést, je obrácení pořadí 2.3.

Tedy

$$M_{i-2} = M_i + f(\mathbf{k_{i-1}}, M_{i-1}), \tag{2.5}$$

$d \geq i \geq 2$, až dospějeme k původní zprávě $M = (M_0, M_1)$.

Za předpokladu, že příjemce zná tvar funkce f a hodnoty šifrovacích klíčů, je dešifrování kryptogramu právě tak těžké jako jeho zašifrování.

Použití NP-těžkých problémů IV

Poznamenejme, že díky (2.5) funkce f nemusí být invertibilní.

Použití NP-těžkých problémů IV

Poznamenejme, že díky (2.5) funkce f nemusí být invertibilní.

V typické situaci, Mr. X může znát tvar funkce f, ale nesmí znát klíče. Zda je pak schopen získat klíče pomocí luštění s pomocí volby, závisí na tvaru f.

Použití NP-těžkých problémů IV

Poznamenejme, že díky (2.5) funkce f nemusí být invertibilní.

V typické situaci, Mr. X může znát tvar funkce f, ale nesmí znát klíče. Zda je pak schopen získat klíče pomocí luštění s pomocí volby, závisí na tvaru f.

Je-li f lineární transformace, je určení klíče výpočetně snadné. Avšak, v případě, že f je nelineární transformace, např.

$$f(x_1, \ldots, x_n; y_1, \ldots, y_n) = (p_1, \ldots, p_n),$$

kde každé p_i je polynom v proměnných

$$(x_1, \ldots, x_n; y_1, \ldots, y_n),$$

pak určení klíče ze znalosti M a C se redukuje na problém vyřešení algebraických rovnic pro klíčové proměnné.

Použití NP-těžkých problémů V

Jak jsme viděli, to je NP-těžký problém.

Použití NP-těžkých problémů V

Jak jsme viděli, to je NP-těžký problém.

Dosáhli jsme tedy našeho cíle: nalezení kryptosystému, který má lehký proces šifrování a dešifrování, ale který zároveň vynucuje po Mr. X, aby při hledání klíče pomocí luštění s pomocí volby řešil NP-těžký problém. Za předpokladu, že NP-těžké problémy jsou nezvládnutelné, je bezpečnost systému zajištěna.

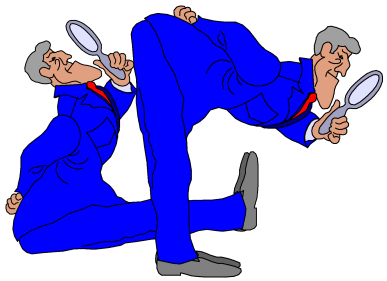
Použití NP-těžkých problémů V

Jak jsme viděli, to je NP-těžký problém.

Dosáhli jsme tedy našeho cíle: nalezení kryptosystému, který má lehký proces šifrování a dešifrování, ale který zároveň vynucuje po Mr. X, aby při hledání klíče pomocí luštění s pomocí volby řešil NP-těžký problém. Za předpokladu, že NP-těžké problémy jsou nezvládnutelné, je bezpečnost systému zajištěna.

Je zde však následující problém. Klasická teorie složitosti se téměř úplně zabývá s nejhorším možným chováním. Ačkoliv tedy určení klíče je těžký problém, nemáme jistotu, že neexistuje velký počet "snadných vstupů".

O čem to bude



Popis vlivu I

Cílem počáteční permutace *IP* je provést rozprostření vlivu bitů z jedněch bajtů otevřeného textu *M* do ostatních.

Popis vlivu I

Cílem počáteční permutace IP je provést rozprostření vlivu bitů z jedněch bajtů otevřeného textu M do ostatních.

Permutace IP^{-1} je zde proto, aby se v dešifrovacím procesu napravil účinek IP. Kryptograficky je tato permutace nevýznamná a v rozborech je zanedbáváná.

Popis vlivu I

Cílem počáteční permutace IP je provést rozprostření vlivu bitů z jedněch bajtů otevřeného textu M do ostatních.

Permutace IP^{-1} je zde proto, aby se v dešifrovacím procesu napravil účinek IP. Kryptograficky je tato permutace nevýznamná a v rozborech je zanedbáváná.

Skutečně, při útoku se znalostí otevřeného textu M, tj. při znalosti odpovídajících si dvojic M otevřeného textu a C šifrového textu, si vliv permutace IP na otevřený i šifrový text snadno eliminujeme. Uvažujeme-li $IP^{-1}(M)$ a $IP(C)$, je to totéž, jako by schéma tyto permutace vůbec neobsahovalo.

Popis vlivu I

Cílem počáteční permutace IP je provést rozprostření vlivu bitů z jedněch bajtů otevřeného textu M do ostatních.

Permutace IP^{-1} je zde proto, aby se v dešifrovacím procesu napravil účinek IP. Kryptograficky je tato permutace nevýznamná a v rozborech je zanedbáváná.

Skutečně, při útoku se znalostí otevřeného textu M, tj. při znalosti odpovídajících si dvojic M otevřeného textu a C šifrového textu, si vliv permutace IP na otevřený i šifrový text snadno eliminujeme. Uvažujeme-li $IP^{-1}(M)$ a $IP(C)$, je to totéž, jako by schéma tyto permutace vůbec neobsahovalo.

Také závěrečná výměna slov L a R je zde jen proto, aby dešifrovací proces byl shodný se šifrovacím, tj. pro možnost zahájení rekonstrukce předchozích dvojic (L, R) z následujících.

Popis vlivu II

Základem schématu je transformace f, která vytváří tzv. ***substitučně-permutační síť***. V podstatě se skládá ze substituce (S-boxy) a permutace P a zároveň zajišťuje vliv klíče na proces zpracování otevřeného textu.

Popis vlivu II

Základem schématu je transformace f, která vytváří tzv. ***substitučně-permutační síť***. V podstatě se skládá ze substituce (S-boxy) a permutace P a zároveň zajišťuje vliv klíče na proces zpracování otevřeného textu.

Klíč je na proměnné R načítán v modulu 2 jako heslo na otevřený text u Vernamovy šifry, permutace P nám připomíná historické transpoziční systémy a S-boxy substituční systémy. DES je pak jejich ***součinová šifra***.

Popis vlivu II

Základem schématu je transformace f, která vytváří tzv. ***substitučně-permutační síť***. V podstatě se skládá ze substituce (S-boxy) a permutace P a zároveň zajišťuje vliv klíče na proces zpracování otevřeného textu.

Klíč je na proměnné R načítán v modulu 2 jako heslo na otevřený text u Vernamovy šifry, permutace P nám připomíná historické transpoziční systémy a S-boxy substituční systémy. DES je pak jejich ***součinová šifra***.

DES jako celek je substituční systémem, pracujícím se slovy délky 64 bitů. Je to tedy kódová kniha, která má 2^{64} kódových výrazů, neboť to je počet všech možných otevřených textů.

Popis vlivu II

Základem schématu je transformace f, která vytváří tzv. ***substitučně-permutační síť***. V podstatě se skládá ze substituce (S-boxy) a permutace P a zároveň zajišťuje vliv klíče na proces zpracování otevřeného textu.

Klíč je na proměnné R načítán v modulu 2 jako heslo na otevřený text u Vernamovy šifry, permutace P nám připomíná historické transpoziční systémy a S-boxy substituční systémy. DES je pak jejich ***součinová šifra***.

DES jako celek je substituční systémem, pracujícím se slovy délky 64 bitů. Je to tedy kódová kniha, která má 2^{64} kódových výrazů, neboť to je počet všech možných otevřených textů.

Kódové výrazy se nevyhledávají, ale na základě znalosti klíče se vypočítávají. Ten, kdo nezná klíč, by měl být postaven před neřešitelnou úlohu "***dekódování***".

Popis vlivu III

Cílem algoritmu je, aby v uvedené kódové knize neexistovala jiná skrytá souvislost mezi klíčem K, otevřeným textem M a šifrovým textem C, která by byla využitelná k luštění.

Popis vlivu III

Cílem algoritmu je, aby v uvedené kódové knize neexistovala jiná skrytá souvislost mezi klíčem K, otevřeným textem M a šifrovým textem C, která by byla využitelná k luštění.

Jednou z vlastností DESu, která se rovněž statisticky testovala, je vliv změny jednoho bitu v otevřeném textu M (resp. v klíči K), aby pravděpodobnost změny každého bitu šifrového textu C byla asi 50%.

Popis vlivu III

Cílem algoritmu je, aby v uvedené kódové knize neexistovala jiná skrytá souvislost mezi klíčem K, otevřeným textem M a šifrovým textem C, která by byla využitelná k luštění.

Jednou z vlastností DESu, která se rovněž statisticky testovala, je vliv změny jednoho bitu v otevřeném textu M (resp. v klíči K), aby pravděpodobnost změny každého bitu šifrového textu C byla asi 50%.

Tímto bude mj. zaručeno, že šifrové výrazy odpovídající dvěma málo se lišícím otevřeným textů budou naprosto odlišné. Podobně to je i s požadavkem vlivu klíče na šifrový text.

Popis vlivu III

Cílem algoritmu je, aby v uvedené kódové knize neexistovala jiná skrytá souvislost mezi klíčem K, otevřeným textem M a šifrovým textem C, která by byla využitelná k luštění.

Jednou z vlastností DESu, která se rovněž statisticky testovala, je vliv změny jednoho bitu v otevřeném textu M (resp. v klíči K), aby pravděpodobnost změny každého bitu šifrového textu C byla asi 50%.

Tímto bude mj. zaručeno, že šifrové výrazy odpovídající dvěma málo se lišícím otevřeným textů budou naprosto odlišné. Podobně to je i s požadavkem vlivu klíče na šifrový text.

Dále se požadovalo, aby neexistovala žádná korelace mezi otevřeným textem M a šifrovým textem C a mezi šifrovým textem C a klíčem K. Tyto vlastnosti byly statisticky potvrzeny a testovány.

Popis vlivu IV

Výběrové statistické testy pak mohou potvrdit jen některé vlastnosti, ale nemohou vyloučit nekonečně mnoho dalších.

To se týká například vlastnosti ***komplementárnosti***, kterou budeme diskutovat dále. Jak na ni máme přijít statistickými testy, když nevíme, že vůbec existuje?

Popis vlivu IV

Výběrové statistické testy pak mohou potvrdit jen některé vlastnosti, ale nemohou vyloučit nekonečně mnoho dalších.

To se týká například vlastnosti ***komplementárnosti***, kterou budeme diskutovat dále. Jak na ni máme přijít statistickými testy, když nevíme, že vůbec existuje?

Dalšími požadovanými vlastnostmi jsou ***konfúze a difúze***.

Popis vlivu IV

Výběrové statistické testy pak mohou potvrdit jen některé vlastnosti, ale nemohou vyloučit nekonečně mnoho dalších.

To se týká například vlastnosti ***komplementárnosti***, kterou budeme diskutovat dále. Jak na ni máme přijít statistickými testy, když nevíme, že vůbec existuje?

Dalšími požadovanými vlastnostmi jsou ***konfúze a difúze***.

Jedná se o to, aby měl každý bit klíče K a otevřeného textu M vliv na každý bit šifrového textu C a aby tento vliv byl velmi komplikovaný.

Popis vlivu IV

Výběrové statistické testy pak mohou potvrdit jen některé vlastnosti, ale nemohou vyloučit nekonečně mnoho dalších.

To se týká například vlastnosti ***komplementárnosti***, kterou budeme diskutovat dále. Jak na ni máme přijít statistickými testy, když nevíme, že vůbec existuje?

Dalšími požadovanými vlastnostmi jsou ***konfúze a difúze***.

Jedná se o to, aby měl každý bit klíče K a otevřeného textu M vliv na každý bit šifrového textu C a aby tento vliv byl velmi komplikovaný.

Na složitost zde pak mají největší vliv S-boxy.

Popis vlivu IV

Výběrové statistické testy pak mohou potvrdit jen některé vlastnosti, ale nemohou vyloučit nekonečně mnoho dalších.

To se týká například vlastnosti ***komplementárnosti***, kterou budeme diskutovat dále. Jak na ni máme přijít statistickými testy, když nevíme, že vůbec existuje?

Dalšími požadovanými vlastnostmi jsou ***konfúze a difúze***.

Jedná se o to, aby měl každý bit klíče K a otevřeného textu M vliv na každý bit šifrového textu C a aby tento vliv byl velmi komplikovaný.

Na složitost zde pak mají největší vliv S-boxy.

Každý výstupní bit S-boxu je nelineární funkcí všech vstupních bitů (při vyjádření operacemi $\oplus$ a $\wedge$).

Popis vlivu V

V případě, že by S-boxy realizovaly lineární funkci, celé schéma by realizovalo pouze lineární funkci.

Popis vlivu V

V případě, že by S-boxy realizovaly lineární funkci, celé schéma by realizovalo pouze lineární funkci.

Potom by ovšem všech 64 bitů šifrového textu C bylo jen různými lineárními kombinacemi bitů otevřeného textu M a klíče K. Ale takovéto schéma bychom mohli okamžitě rozluštit řešením soustavy lineárních rovnic.

Popis vlivu V

V případě, že by S-boxy realizovaly lineární funkci, celé schéma by realizovalo pouze lineární funkci.

Potom by ovšem všech 64 bitů šifrového textu C bylo jen různými lineárními kombinacemi bitů otevřeného textu M a klíče K. Ale takovéto schéma bychom mohli okamžitě rozluštit řešením soustavy lineárních rovnic.

Nelineárním vlastnostem se při studiu DES věnovala velká pozornost, neboť zajišťují požadovanou konfúzi. Bohužel, kritéria návrhu S-boxů zůstávají doposud tajná a přitom právě zde bylo nalezeno mnoho slabin.

Popis vlivu V

V případě, že by S-boxy realizovaly lineární funkci, celé schéma by realizovalo pouze lineární funkci.

Potom by ovšem všech 64 bitů šifrového textu C bylo jen různými lineárními kombinacemi bitů otevřeného textu M a klíče K. Ale takovéto schéma bychom mohli okamžitě rozluštit řešením soustavy lineárních rovnic.

Nelineárním vlastnostem se při studiu DES věnovala velká pozornost, neboť zajišťují požadovanou konfúzi. Bohužel, kritéria návrhu S-boxů zůstávají doposud tajná a přitom právě zde bylo nalezeno mnoho slabin.

U DES byly stanoveny ***4 základní módy činnosti***, které byly odvozeny od různých potřeb. Základem je vlastní blokový algoritmus, tj. zobrazení $E_K : X \to X$, kde $X = \{0,1\}^{\{1,2,\ldots,64\}}$ je množina všech 64-bitových bloků.

Režimy činnosti DES I

Tento algoritmus je zároveň prvním módem.

1. **ECB** – Elektronická kódová kniha, píšeme

$$C = E_K(M), \tag{3.1}$$

kde M je otevřený text a C je šifrový text. Jedná se skutečně o kódovou knihu, neboť při opakovaných blocích otevřeného textu M jsou šifrové zprávy C stejné. Proto by bylo možné při šifrování zpráv tímto módem z operativních informací domýšlet část zašifrovaného obsahu bez nutnosti jejich luštění.

Režimy činnosti DES I

Tento algoritmus je zároveň prvním módem.

1. **ECB** – Elektronická kódová kniha, píšeme

$$C = E_K(M), \tag{3.1}$$

kde M je otevřený text a C je šifrový text. Jedná se skutečně o kódovou knihu, neboť při opakovaných blocích otevřeného textu M jsou šifrové zprávy C stejné. Proto by bylo možné při šifrování zpráv tímto módem z operativních informací domýšlet část zašifrovaného obsahu bez nutnosti jejich luštění.
Formalizované části zpráv by bylo možné opakovat v zašifrovaném tvaru (platební příkazy, standardní hlášení). Proto se tento mód nepoužívá k šifrování zpráv, ale jen k šifrování klíčů.

Režimy činnosti DES II

2. **CBC** – Zřetězení šifrového textu, symbolicky zapsáno jako

$$C_n = E_K(M_n \oplus C_{n-1}), \tag{3.2}$$

kde $(M_n)_n$ je systém bloků otevřeného textu.

Tento mód využívá výstup z jednoho kroku šifrování k modifikaci nového vstupu, takže každý blok C_n šifrového textu je závislý nejen na právě šifrovaném bloku M_n otevřeného textu, ale i na všech předchozích blocích otevřeného textu a šifrového textu.

Režimy činnosti DES III

3. **CFB** – Zpětná vazba ze šifrového textu, píšeme

$$C_n = M_n \oplus E_K(I_n), \tag{3.3}$$

kde I_n je vstupní registr posunutý o k bitů doleva a doplněný zprava k bity C_{n-1}.

Režimy činnosti DES III

3. **CFB** – Zpětná vazba ze šifrového textu, píšeme

$$C_n = M_n \oplus E_K(I_n), \tag{3.3}$$

kde I_n je vstupní registr posunutý o k bitů doleva a doplněný zprava k bity C_{n-1}.

Přitom M_n je proud $k-$bitových znaků otevřeného textu. Je to mód vhodný tam, kde se data zpracovávají po znacích nebo po částech menších než 64 bitů. Zpětná vazba je vedena ze šifrového textu, ale jen v délce k bitů, přičemž u výstupu z blokového algoritmu je využíváno také jen k bitů.

Režimy činnosti DES III

3. **CFB** – Zpětná vazba ze šifrového textu, píšeme

$$C_n = M_n \oplus E_K(I_n), \tag{3.3}$$

kde I_n je vstupní registr posunutý o k bitů doleva a doplněný zprava k bity C_{n-1}.

Přitom M_n je proud $k-$bitových znaků otevřeného textu. Je to mód vhodný tam, kde se data zpracovávají po znacích nebo po částech menších než 64 bitů. Zpětná vazba je vedena ze šifrového textu, ale jen v délce k bitů, přičemž u výstupu z blokového algoritmu je využíváno také jen k bitů.

Je to systém podobný proudové šifře, avšak zachovávající vlastnost závislosti šifrového textu na předchozím otevřeném textu a šifrovém textu.

Režimy činnosti DES IV

4. **OFB** – Zpětná vazba z výstupu, symbolicky zapsáno jako

$$H_n = E_K(J_n), \qquad C_n = M_n \oplus H_n, \tag{3.4}$$

kde J_n je vstupní registr posunutý o k bitů doleva a doplněný zprava k bity H_{n-1}, M_n je proud k-bitových znaků otevřeného textu a H_n je vytvořený proud hesla.

Režimy činnosti DES IV

4. **OFB** – Zpětná vazba z výstupu, symbolicky zapsáno jako

$$H_n = E_K(J_n), \qquad C_n = M_n \oplus H_n, \tag{3.4}$$

kde J_n je vstupní registr posunutý o k bitů doleva a doplněný zprava k bity H_{n-1}, M_n je proud k-bitových znaků otevřeného textu a H_n je vytvořený proud hesla.

Tento mód pracuje podobně jako Vernamova šifra, pouze zdroj hesla není náhodný. Mód byl navržen pro potřeby, kdy je nežádoucí, aby se chyby vzniklé na komunikačním kanálu rozšiřovaly působením šifrového algoritmu do otevřeného textu.

Režimy činnosti DES IV

4. **OFB** – Zpětná vazba z výstupu, symbolicky zapsáno jako

$$H_n = E_K(J_n), \qquad C_n = M_n \oplus H_n, \tag{3.4}$$

kde J_n je vstupní registr posunutý o k bitů doleva a doplněný zprava k bity H_{n-1}, M_n je proud k-bitových znaků otevřeného textu a H_n je vytvořený proud hesla.

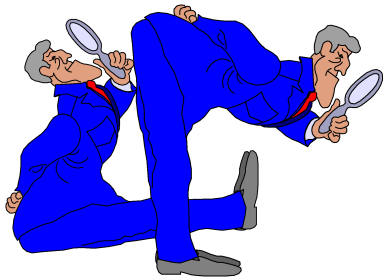
Tento mód pracuje podobně jako Vernamova šifra, pouze zdroj hesla není náhodný. Mód byl navržen pro potřeby, kdy je nežádoucí, aby se chyby vzniklé na komunikačním kanálu rozšiřovaly působením šifrového algoritmu do otevřeného textu.

Jde například o přenosy dat s vysokou rychlostí a redundancí (kódovaná řeč, videosignál, satelitní spoje aj.).

Režimy činnosti DES V

Poznamenejme, že u módů CBC, CBF a OFB je nutné "***vstupní registr***" nejprve na začátku naplnit nějakou hodnotou. Ta se nazývá ***inicializační vektor*** a odesílá se většinou na začátku zprávy.

O čem to bude



- Příliš krátký klíč
- Pracovní konference NBS
- Komplementárnost a pracovní bloky
- Druhá konference NBS

Příliš krátký klíč I

Prvními velkými kritiky návrhu DES se stali ***Diffie a Hellman*** ze Stanfordské univerzity.

Příliš krátký klíč I

Prvními velkými kritiky návrhu DES se stali ***Diffie a Hellman*** ze Stanfordské univerzity.

Ve svém dopise NBS nejvíce kritizovali ***malý objem klíče – 56 bitů***. Poukazovali na to, že nehledě na jiný možný útok k rozbití DES postačí pouhé zkoušení možných klíčů.

Příliš krátký klíč I

Prvními velkými kritiky návrhu DES se stali ***Diffie a Hellman*** ze Stanfordské univerzity.

Ve svém dopise NBS nejvíce kritizovali ***malý objem klíče – 56 bitů***. Poukazovali na to, že nehledě na jiný možný útok k rozbití DES postačí pouhé zkoušení možných klíčů.

To může provádět kdokoli, kdykoli a se zaručeným úspěchem. Dokazovali, že to bylo možné i s tehdejší technologií (1975).

Příliš krátký klíč I

Prvními velkými kritiky návrhu DES se stali ***Diffie a Hellman*** ze Stanfordské univerzity.

Ve svém dopise NBS nejvíce kritizovali ***malý objem klíče – 56 bitů***. Poukazovali na to, že nehledě na jiný možný útok k rozbití DES postačí pouhé zkoušení možných klíčů.

To může provádět kdokoli, kdykoli a se zaručeným úspěchem. Dokazovali, že to bylo možné i s tehdejší technologií (1975).

Uvažovali o sestrojení ***speciálního stroje***, který by pro danou dvojici otevřený text M a šifrový klíč C hledal použitý klíč vyzkoušením všech jeho možností.

Příliš krátký klíč I

Prvními velkými kritiky návrhu DES se stali ***Diffie a Hellman*** ze Stanfordské univerzity.

Ve svém dopise NBS nejvíce kritizovali ***malý objem klíče – 56 bitů***. Poukazovali na to, že nehledě na jiný možný útok k rozbití DES postačí pouhé zkoušení možných klíčů.

To může provádět kdokoli, kdykoli a se zaručeným úspěchem. Dokazovali, že to bylo možné i s tehdejší technologií (1975).

Uvažovali o sestrojení ***speciálního stroje***, který by pro danou dvojici otevřený text M a šifrový klíč C hledal použitý klíč vyzkoušením všech jeho možností.

V době kritiky, tj. v říjnu 1975, odhadovali, že by s tímto strojem vyluštili za jeden den jednu takovou úlohu ***za cenu 10 000 $***.

Příliš krátký klíč II

Speciální stroj by potřeboval vyzkoušet 2^{56} tj. cca. 10^{17} klíčů za jeden den, což je přibližně 10^{12} klíčů za sekundu.

Příliš krátký klíč II

Speciální stroj by potřeboval vyzkoušet 2^{56} tj. cca. 10^{17} klíčů za jeden den, což je přibližně 10^{12} klíčů za sekundu.

Při ceně 10 $ za čip by tento stroj stál ***20 000 000 $***
(10 000 000 $ na čipy a zbytek na ostatní náklady).

Příliš krátký klíč II

Speciální stroj by potřeboval vyzkoušet 2^{56} tj. cca. 10^{17} klíčů za jeden den, což je přibližně 10^{12} klíčů za sekundu.

Při ceně 10 $ za čip by tento stroj stál ***20 000 000 $***
(10 000 000 $ na čipy a zbytek na ostatní náklady).

Při úplném odepsání stroje za 5 let by tak denní provoz stál 10 000 $. Ve skutečnosti se s padesátiprocentní pravděpodobností narazí na správný klíč už po vyzkoušení poloviny klíčů.

Příliš krátký klíč II

Speciální stroj by potřeboval vyzkoušet 2^{56} tj. cca. 10^{17} klíčů za jeden den, což je přibližně 10^{12} klíčů za sekundu.

Při ceně 10 $ za čip by tento stroj stál ***20 000 000 $*** (10 000 000 $ na čipy a zbytek na ostatní náklady).

Při úplném odepsání stroje za 5 let by tak denní provoz stál 10 000 $. Ve skutečnosti se s padesátiprocentní pravděpodobností narazí na správný klíč už po vyzkoušení poloviny klíčů.

To by hledání klíče dvakrát zrychlilo a tím i o polovinu snížilo náklady. Dále argumentovali tím, že během 5 let se cena technologie sníží v průměru 10-krát, a proto by za 10 let tento stroj stál pouze ***200 000 $***.

Příliš krátký klíč II

Speciální stroj by potřeboval vyzkoušet 2^{56} tj. cca. 10^{17} klíčů za jeden den, což je přibližně 10^{12} klíčů za sekundu.

Při ceně 10 $ za čip by tento stroj stál ***20 000 000 $*** (10 000 000 $ na čipy a zbytek na ostatní náklady).

Při úplném odepsání stroje za 5 let by tak denní provoz stál 10 000 $. Ve skutečnosti se s padesátiprocentní pravděpodobností narazí na správný klíč už po vyzkoušení poloviny klíčů.

To by hledání klíče dvakrát zrychlilo a tím i o polovinu snížilo náklady. Dále argumentovali tím, že během 5 let se cena technologie sníží v průměru 10-krát, a proto by za 10 let tento stroj stál pouze ***200 000 $***.

Přitom si v odhadu nedělali nárok na přesnost v rozmezí jednoho řádu!

Příliš krátký klíč III

Doporučovali rozšířit klíč na 128 nebo 256 bitů nebo šifrovat víckrát za sebou s použitím nezávislých klíčů, tedy šifrový text

$$C = E_{K_1}(E_{K_2}(\ldots E_{K_m}(M)\ldots)).$$

Pracovní konference NBS I

NBS reagoval na jejich argumenty svoláním pracovní konference, která se konala 30.-31. srpna 1976 a byla zaměřena na to, zda předpovědi a tvrzení Diffieho a Hellmana jsou realistické.

Pracovní konference NBS I

NBS reagoval na jejich argumenty svoláním pracovní konference, která se konala 30.-31. srpna 1976 a byla zaměřena na to, zda předpovědi a tvrzení Diffieho a Hellmana jsou realistické.

Závěr z konference byl, že tehdejší technologií by jejich stroj nebylo možné sestrojit, a pokud se ho sestrojit podaří, nebude to dříve než po roce 1990 (a to asi za 72 000 000 $).

Pracovní konference NBS I

NBS reagoval na jejich argumenty svoláním pracovní konference, která se konala 30.-31. srpna 1976 a byla zaměřena na to, zda předpovědi a tvrzení Diffieho a Hellmana jsou realistické.

Závěr z konference byl, že tehdejší technologií by jejich stroj nebylo možné sestrojit, a pokud se ho sestrojit podaří, nebude to dříve než po roce 1990 (a to asi za 72 000 000 $).

Proti zvětšení klíče bylo argumentováno tím, že by to vedlo k prodražení čipů, horším podmínkám vývozu a že to není nezbytné ani pro zamýšlené aplikace, ani pro uvažovanou životnost schématu (10-15 let).

Pracovní konference NBS II

Diffie a Hellman na to odpověděli článkem "***Exhaustive Cryptoanalysis of the NBS Data Encryption Standard***" (1977), v němž vyčerpávajícím způsobem a podloženými argumenty dokazovali, že jejich odhady jsou správné (***dnes víme, že měli pravdu***).

Pracovní konference NBS II

Diffie a Hellman na to odpověděli článkem "***Exhaustive Cryptoanalysis of the NBS Data Encryption Standard***" (1977), v němž vyčerpávajícím způsobem a podloženými argumenty dokazovali, že jejich odhady jsou správné (***dnes víme, že měli pravdu***).

Mezitím i interní studie IBM odhadla, že by takový stroj mohl být sestrojen do roku 1981 a za cenu 200 000 000 $, tedy v rámci jejich tolerance.

Pracovní konference NBS II

Diffie a Hellman na to odpověděli článkem "***Exhaustive Cryptoanalysis of the NBS Data Encryption Standard***" (1977), v němž vyčerpávajícím způsobem a podloženými argumenty dokazovali, že jejich odhady jsou správné (***dnes víme, že měli pravdu***).

Mezitím i interní studie IBM odhadla, že by takový stroj mohl být sestrojen do roku 1981 a za cenu 200 000 000 $, tedy v rámci jejich tolerance.

V roce 1997 byla agenturou RSA vypsána kryptoanalytická soutěž s cílem prolomit šifru DES se znalostí začátku otevřeného textu a neznámým klíčem o délce 56 bitů.

Pracovní konference NBS II

Diffie a Hellman na to odpověděli článkem "***Exhaustive Cryptoanalysis of the NBS Data Encryption Standard***" (1977), v němž vyčerpávajícím způsobem a podloženými argumenty dokazovali, že jejich odhady jsou správné (***dnes víme, že měli pravdu***).

Mezitím i interní studie IBM odhadla, že by takový stroj mohl být sestrojen do roku 1981 a za cenu 200 000 000 $, tedy v rámci jejich tolerance.

V roce 1997 byla agenturou RSA vypsána kryptoanalytická soutěž s cílem prolomit šifru DES se znalostí začátku otevřeného textu a neznámým klíčem o délce 56 bitů.

Vítězem se stal řešitelský tým ***DES challenge*** pod vedením ***R. Verbena***.

Pracovní konference NBS III

Využil distribuovaného výpočtu a postupu, který definovali Diffie a Hellman o 24 let dříve.

Pracovní konference NBS III

Využil distribuovaného výpočtu a postupu, který definovali Diffie a Hellman o 24 let dříve.

S výpočetním výkonem roku 1999 stačilo k prolomení šifry méně než 24 hodin.

Pracovní konference NBS III

Využil distribuovaného výpočtu a postupu, který definovali Diffie a Hellman o 24 let dříve.

S výpočetním výkonem roku 1999 stačilo k prolomení šifry méně než 24 hodin.

Na konci roku 1999 bylo doporučeno, aby se přešlo z DES na její vylepšenou verzi 3DES triple DES; trojitý DES).

Pracovní konference NBS III

Využil distribuovaného výpočtu a postupu, který definovali Diffie a Hellman o 24 let dříve.

S výpočetním výkonem roku 1999 stačilo k prolomení šifry méně než 24 hodin.

Na konci roku 1999 bylo doporučeno, aby se přešlo z DES na její vylepšenou verzi 3DES triple DES; trojitý DES).

3DES je trojnásobná aplikace klíčem. Klíč šifry 3DES je tedy 168 bitů dlouhý.

Komplementárnost a pracovní bloky I

Protože Diffie, Hellman a dalších pět jejich kolegů nesouhlasili s tím, že IBM a NSA utajují výsledky svého zkoumání bezpečnosti DES a návrhová kritéria, zorganizovali během srpna 1976 vlastní krátké studium DESu "***Results of an initial attempt to cryptoanalyze the NBS Data Encryption Standard***" (1976).

Komplementárnost a pracovní bloky I

Protože Diffie, Hellman a dalších pět jejich kolegů nesouhlasili s tím, že IBM a NSA utajují výsledky svého zkoumání bezpečnosti DES a návrhová kritéria, zorganizovali během srpna 1976 vlastní krátké studium DESu "***Results of an initial attempt to cryptoanalyze the NBS Data Encryption Standard***" (1976).

Z něho vyplynuly další závažné slabosti DES: vlastnost komplementárnosti, určité pravidelnosti v S-boxech a jejich dostatečná nelinearita.

Komplementárnost a pracovní bloky I

Protože Diffie, Hellman a dalších pět jejich kolegů nesouhlasili s tím, že IBM a NSA utajují výsledky svého zkoumání bezpečnosti DES a návrhová kritéria, zorganizovali během srpna 1976 vlastní krátké studium DESu "***Results of an initial attempt to cryptoanalyze the NBS Data Encryption Standard***" (1976).

Z něho vyplynuly další závažné slabosti DES: vlastnost komplementárnosti, určité pravidelnosti v S-boxech a jejich dostatečná nelinearita.

Poukazovali na to, že veřejný standard by měl mít i ***veřejná návrhová kritéria***. V opačném případě to budí nedůvěru. Těm, kdo návrhová kritéria znají, umožňuje využít jejich slabosti k luštění, a jsou tedy proti ostatním zvýhodněni.

Komplementárnost a pracovní bloky II

Mohlo by jít buď o osoby, které se k těmto tajným informacím dostanou neoprávněně (např. cizí tajné služby), nebo o samotné ochránce (IBM, NSA). To by u veřejného standardu nemělo být.

Komplementárnost a pracovní bloky II

Mohlo by jít buď o osoby, které se k těmto tajným informacím dostanou neoprávněně (např. cizí tajné služby), nebo o samotné ochránce (IBM, NSA). To by u veřejného standardu nemělo být.

Účastníci studia se domnívali, že NSA si ve skutečnosti nepřeje silný standard, aby jej mohla snadněji luštit.

Komplementárnost a pracovní bloky II

Mohlo by jít buď o osoby, které se k těmto tajným informacím dostanou neoprávněně (např. cizí tajné služby), nebo o samotné ochránce (IBM, NSA). To by u veřejného standardu nemělo být.

Účastníci studia se domnívali, že NSA si ve skutečnosti nepřeje silný standard, aby jej mohla snadněji luštit.

Nesouhlasili s tím, že by DES měl být odolný jen proti "***luštění se znalostí otevřeného textu***", což NBS do požadavků dal, ale i proti "***luštění s možností volby otevřeného textu***", což původně požadováno nebylo.

Druhá konference NBS I

Nato NBS zorganizoval pracovní setkání matematiků v září 1976 k analýze kvality DES a možnosti, že by mohl obsahovat "trojského koně" ("***Report of the workshop on cryptography on support of computer security***").

Druhá konference NBS I

Nato NBS zorganizoval pracovní setkání matematiků v září 1976 k analýze kvality DES a možnosti, že by mohl obsahovat "trojského koně" ("***Report of the workshop on cryptography on support of computer security***").

Přestože jiní účastníci na různé nedostatky také upozorňovali, fakticky nebyla předložena žádná konkrétní metoda na jejich přímé využití. Představitel IBM na konferenci k problému "trojského koně" dokonce prohlásil: "**Musíte nám důvěřovat, my všichni jsme dobří skauti**". Také NBS a NSA prohlásily, že neznají žádné metody, jak využít objevené kvazilinearity.

Druhá konference NBS I

Nato NBS zorganizoval pracovní setkání matematiků v září 1976 k analýze kvality DES a možnosti, že by mohl obsahovat "trojského koně" ("***Report of the workshop on cryptography on support of computer security***").

Přestože jiní účastníci na různé nedostatky také upozorňovali, fakticky nebyla předložena žádná konkrétní metoda na jejich přímé využití. Představitel IBM na konferenci k problému "trojského koně" dokonce prohlásil: "**Musíte nám důvěřovat, my všichni jsme dobří skauti**". Také NBS a NSA prohlásily, že neznají žádné metody, jak využít objevené kvazilinearity.

Bezprostředně po skončení konferencí se NBS rozhodl ponechat DES beze změn. Toto rozhodnutí způsobil zejména tlak průmyslu, který už potřeboval konečný verdikt, aby mohl vyrábět čipy.

Druhá konference NBS II

Změny v DES by tento proces nejméně o rok oddálily. Aféra kolem utajovaní však zanechala na NSA nesmazatelný stín podezření.

Druhá konference NBS II

Změny v DES by tento proces nejméně o rok oddálily. Aféra kolem utajovaní však zanechala na NSA nesmazatelný stín podezření.

Zvláště když vyšetřovací komise Senátu USA potvrdila, že NSA přesvědčila IBM o vhodnosti redukce délky klíče. Původně totiž IBM pro své potřeby používala 128-bitový klíč.

Druhá konference NBS II

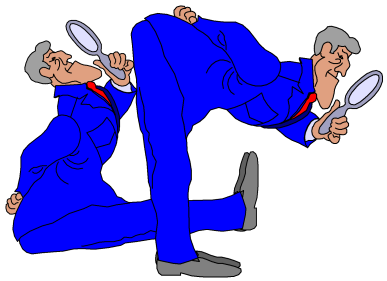
Změny v DES by tento proces nejméně o rok oddálily. Aféra kolem utajovaní však zanechala na NSA nesmazatelný stín podezření.

Zvláště když vyšetřovací komise Senátu USA potvrdila, že NSA přesvědčila IBM o vhodnosti redukce délky klíče. Původně totiž IBM pro své potřeby používala 128-bitový klíč.

V roce 1978 prohlásil Davis na podporu DES následující:

> "***Každý, kdo si zakoupí kryptografické zařízení pracující na základě DES, může být ujištěn o specifické úrovni bezpečnosti dat: totiž je potřeba použít*** 2^{55} ***pokusů a metodu prověření všech možností, aby bylo možno získat klíč pro použitý šifrovací algoritmus.***"

O čem to bude



Komplementárnost I

Objevená vlastnost komplementárnosti spočívá v tom, že ze vztahu

$$C = E_K(M) \quad \text{plyne} \quad \text{non } C = E_{\text{non } K}(\text{non } M), \tag{5.1}$$

kde non označuje negaci bit po bitu.

Komplementárnost I

Objevená vlastnost komplementárnosti spočívá v tom, že ze vztahu

$$C = E_K(M) \quad \text{plyne} \quad \text{non } C = E_{\text{non } K}(\text{non } M), \tag{5.1}$$

kde non označuje negaci bit po bitu.

To je pravidelnost, která by se v takovémto případě rozhodně neměla objevit. Autoři rozboru její odhalení považovali diplomaticky řečeno za velmi znepokojující.

Komplementárnost I

Objevená vlastnost komplementárnosti spočívá v tom, že ze vztahu

$$C = E_K(M) \quad \text{plyne} \quad \text{non } C = E_{\text{non } K}(\text{non } M), \tag{5.1}$$

kde non označuje negaci bit po bitu.

To je pravidelnost, která by se v takovémto případě rozhodně neměla objevit. Autoři rozboru její odhalení považovali diplomaticky řečeno za velmi znepokojující.

Tvůrcům schématu tato vlastnost pravděpodobně unikla. Je umožněna tím, že negace klíče i vstupu se při jejich součtu mod 2 ve funkci f zruší:

$$f(R, K) = f(\text{non } R, \text{non } K). \tag{5.2}$$

Komplementárnost II

Totiž označíme-li $R'_0 = \text{non } R_0$, $L'_0 = \text{non } L_0$ a šifrujeme-li zprávu $M' = (L'_0, R'_0)$ klíčem $K' = \text{non } K$ obdržíme postupně:
$M' = \text{non } M$.

Komplementárnost II

Totiž označíme-li $R'_0 = \text{non } R_0$, $L'_0 = \text{non } L_0$ a šifrujeme-li zprávu $M' = (L'_0, R'_0)$ klíčem $K' = \text{non } K$ obdržíme postupně:
$M' = \text{non } M$.

Pokud $(L'_i, R'_i) = \text{non } (L_i, R_i)$, pak
$(L'_{i+1}, R'_{i+1}) = \text{non } (L_{i+1}, R_{+1}i)$, protože

$$\begin{aligned}(L'_{i+1}, R'_{i+1}) &= (R'_i, L'_i \oplus f(R'_i, K'_i)) \\ &= (\text{non } R_i, (\text{non } L_i) \oplus f(\text{non } R_i, \text{non } K_i)) \\ &= (\text{non } R_i, (\text{non } L_i) \oplus f(R_i, K_i)) = (\text{non } R_i, \text{non } R_{i+1}) \\ &= \text{non } (L_{i+1}, R_{+1}i).\end{aligned}$$

Speciálně tedy $(L'_{16}, R'_{16}) = \text{non } (L_{16}, R_{16})$ tj.
$\text{non } E_K(M) = E_{\text{non } K}(\text{non } M)$.

Komplementárnost II

Totiž označíme-li $R'_0 = \text{non } R_0$, $L'_0 = \text{non } L_0$ a šifrujeme-li zprávu $M' = (L'_0, R'_0)$ klíčem $K' = \text{non } K$ obdržíme postupně:
$M' = \text{non } M$.

Pokud $(L'_i, R'_i) = \text{non } (L_i, R_i)$, pak
$(L'_{i+1}, R'_{i+1}) = \text{non } (L_{i+1}, R_{+1}i)$, protože

$$\begin{aligned}(L'_{i+1}, R'_{i+1}) &= (R'_i, L'_i \oplus f(R'_i, K'_i)) \\ &= (\text{non } R_i, (\text{non } L_i) \oplus f(\text{non } R_i, \text{non } K_i)) \\ &= (\text{non } R_i, (\text{non } L_i) \oplus f(R_i, K_i)) = (\text{non } R_i, \text{non } R_{i+1}) \\ &= \text{non } (L_{i+1}, R_{+1}i).\end{aligned}$$

Speciálně tedy $(L'_{16}, R'_{16}) = \text{non } (L_{16}, R_{16})$ tj.
$\text{non } E_K(M) = E_{\text{non } K}(\text{non } M)$.

To se dá pak využít i k luštění v případě, že hledáme klíč K a máme k dispozici dvojice (M, C_1) a $(\text{non } M, C_2)$, které vznikly při šifrování tímto neznámým klíčem.

Komplementárnost III

Proveďme zašifrování $E_K(M)$. Není-li tento výraz roven C_1, vylučujeme klíč K. Kdyby byl při šifrování $\mathrm{non}\ M$ použít klíč $\mathrm{non}\ K$, obdrželi bychom

$$C_2 = E_{\mathrm{non}\ K}(\mathrm{non}\ M) = \mathrm{non}\ E_K(M). \tag{5.3}$$

Komplementárnost III

Proveďme zašifrování $E_K(M)$. Není-li tento výraz roven C_1, vylučujeme klíč K. Kdyby byl při šifrování $\text{non}\ M$ použít klíč $\text{non}\ K$, obdrželi bychom

$$C_2 = E_{\text{non}\ K}(\text{non}\ M) = \text{non}\ E_K(M). \tag{5.3}$$

Nejsou-li si oba krajní výrazy, které máme k dispozici, rovny, můžeme vyloučit i klíč $\text{non}\ K$.

Komplementárnost III

Proveďme zašifrování $E_K(M)$. Není-li tento výraz roven C_1, vylučujeme klíč K. Kdyby byl při šifrování $\mathrm{non}\ M$ použít klíč $\mathrm{non}\ K$, obdrželi bychom

$$C_2 = E_{\mathrm{non}\ K}(\mathrm{non}\ M) = \mathrm{non}\ E_K(M). \tag{5.3}$$

Nejsou-li si oba krajní výrazy, které máme k dispozici, rovny, můžeme vyloučit i klíč $\mathrm{non}\ K$.

Tím jsme nahradili jedno šifrování (klíčem $\mathrm{non}\ K$) za pouhé porovnávání výrazů, což je proti šifrování časově zanedbatelné. Prostor klíčů je tedy de facto poloviční a hledání 2-krát rychlejší.

Komplementárnost III

Proveďme zašifrování $E_K(M)$. Není-li tento výraz roven C_1, vylučujeme klíč K. Kdyby byl při šifrování $\text{non}\ M$ použít klíč $\text{non}\ K$, obdrželi bychom

$$C_2 = E_{\text{non}\ K}(\text{non}\ M) = \text{non}\ E_K(M). \tag{5.3}$$

Nejsou-li si oba krajní výrazy, které máme k dispozici, rovny, můžeme vyloučit i klíč $\text{non}\ K$.

Tím jsme nahradili jedno šifrování (klíčem $\text{non}\ K$) za pouhé porovnávání výrazů, což je proti šifrování časově zanedbatelné. Prostor klíčů je tedy de facto poloviční a hledání 2-krát rychlejší.

Vlastnost komplementarity by navíc mohla ve špatně navržených protokolech způsobit i nežádoucí odhalení chráněných informací. Při všech možných aplikacích na ni musí být dáván pozor.

Komplementárnost IV

Je smutné, že mnozí kryptologové (zejména z NBS) tuto vlastnost nepovažují za podstatnou. Ostatně v oficiálním popisu DES vypracovaném NBS se také tvrdí, že klíč je 64-bitový.

Komplementárnost IV

Je smutné, že mnozí kryptologové (zejména z NBS) tuto vlastnost nepovažují za podstatnou. Ostatně v oficiálním popisu DES vypracovaném NBS se také tvrdí, že klíč je 64-bitový.

Rozhodně nejde o chybu, ale o vytvoření dojmu, že tomu tak je.

Komplementárnost IV

Je smutné, že mnozí kryptologové (zejména z NBS) tuto vlastnost nepovažují za podstatnou. Ostatně v oficiálním popisu DES vypracovaném NBS se také tvrdí, že klíč je 64-bitový.

Rozhodně nejde o chybu, ale o vytvoření dojmu, že tomu tak je.

S "***dobrými skauty***" tedy není asi všechno v pořádku.

Komplementárnost IV

Je smutné, že mnozí kryptologové (zejména z NBS) tuto vlastnost nepovažují za podstatnou. Ostatně v oficiálním popisu DES vypracovaném NBS se také tvrdí, že klíč je 64-bitový.

Rozhodně nejde o chybu, ale o vytvoření dojmu, že tomu tak je.

S "***dobrými skauty***" tedy není asi všechno v pořádku.

Kromě tajných návrhových kritérií, která mohla obsahovat další skryté vady, v studiu bylo poukázáno na ***velkou korelovanost a nedostatečnou nelinearitu*** výstupních bitů S-boxů.

Komplementárnost IV

Je smutné, že mnozí kryptologové (zejména z NBS) tuto vlastnost nepovažují za podstatnou. Ostatně v oficiálním popisu DES vypracovaném NBS se také tvrdí, že klíč je 64-bitový.

Rozhodně nejde o chybu, ale o vytvoření dojmu, že tomu tak je.

S "***dobrými skauty***" tedy není asi všechno v pořádku.

Kromě tajných návrhových kritérií, která mohla obsahovat další skryté vady, v studiu bylo poukázáno na ***velkou korelovanost a nedostatečnou nelinearitu*** výstupních bitů S-boxů.

Například box S4 má pětasedmdesátiprocentní nadbytečnost.

Nevhodný návrh S-boxů I

S-box S4				(x_1, x_6)													
(x_2, x_3, x_4, x_5)				**0 0**				**0 1**				**1 0**				**1 1**			
0	0	0	0	0	1	1	1	1	1	0	1	1	0	1	0	0	0	1	1
0	0	0	1	1	1	0	1	1	0	0	0	0	1	1	0	1	1	1	1
0	0	1	0	1	1	1	0	1	0	1	1	1	0	0	1	0	0	0	0
0	0	1	1	0	0	1	1	0	1	0	1	0	0	0	0	0	1	1	0
0	1	0	0	0	0	0	0	0	1	1	0	1	1	0	0	1	0	1	0
0	1	0	1	0	1	1	0	1	1	1	1	1	0	1	1	0	0	0	0
0	1	1	0	1	0	0	1	0	0	0	0	0	1	1	1	1	1	0	1
0	1	1	1	1	0	1	0	0	0	1	1	1	1	0	1	1	0	0	0
1	0	0	0	0	0	0	1	0	1	0	0	1	1	1	1	1	0	0	1
1	0	0	1	0	0	1	0	0	1	1	1	0	0	0	1	0	1	0	0
1	0	1	0	1	0	0	0	0	0	1	0	0	0	1	1	0	1	0	1
1	0	0	1	0	1	0	1	1	1	0	0	1	1	1	0	1	0	1	1
1	1	0	0	1	0	1	1	0	0	0	1	0	1	0	1	1	1	0	0
1	1	0	1	1	1	0	0	1	0	1	0	0	0	1	0	0	1	1	1
1	1	1	0	0	1	0	0	1	1	1	0	1	0	0	0	0	0	1	0
1	1	1	1	1	1	1	1	1	0	0	1	0	1	0	1	1	1	1	0

Nevhodný návrh S-boxů II

Jeho 4 menší boxy 4×4 (při hodnotách krajních bitů $x_1 x_6 = 00, 01, 10, 11$) jsou snadno odvoditelné pouze od prvního z nich (00).

Nevhodný návrh S-boxů II

Jeho 4 menší boxy 4×4 (při hodnotách krajních bitů $x_1x_6 = 00, 01, 10, 11$) jsou snadno odvoditelné pouze od prvního z nich (00).

Platí dokonce vztah

$$S4(x \oplus 000001) = (12)(34)S4(x) \oplus (x_6, \text{non } x_6, \text{non } x_6, x_6), \tag{5.4}$$

Nevhodný návrh S-boxů II

Jeho 4 menší boxy 4×4 (při hodnotách krajních bitů $x_1x_6 = 00, 01, 10, 11$) jsou snadno odvoditelné pouze od prvního z nich (00).

Platí dokonce vztah

$$S4(x \oplus 000001) = (12)(34)S4(x) \oplus (x_6, \text{non } x_6, \text{non } x_6, x_6), \tag{5.4}$$

kde $x = (x_1, x_2, x_3, x_4, x_5, x_6)$ je vstup a symbolický zápis (1,2) znamená výměnu 1. a 2. bitu.

Nevhodný návrh S-boxů II

Jeho 4 menší boxy 4×4 (při hodnotách krajních bitů $x_1x_6 = 00, 01, 10, 11$) jsou snadno odvoditelné pouze od prvního z nich (00).

Platí dokonce vztah

$$S4(x \oplus 000001) = (12)(34)S4(x) \oplus (x_6, \text{non } x_6, \text{non } x_6, x_6), \tag{5.4}$$

kde $x = (x_1, x_2, x_3, x_4, x_5, x_6)$ je vstup a symbolický zápis (1,2) znamená výměnu 1. a 2. bitu.

Označíme-li $y = (y_1, y_2, y_3, y_4)$ jako výstup, vyplývá odud, že součet $(y_1 \oplus y_2)$ je na x_6 závislý pouze lineárně a $(y_1 \oplus y_2 \oplus y_3 \oplus y_4)$ na něm nezávisí vůbec. Taková pravděpodobnost je nežádoucí.

Slabé a poloslabé klíče

Hodně pozornosti bylo věnováno studiu klíčů. Budou-li registry C a D na počátku obsahovat konstantní bity (buď nuly nebo jedničky), pak je operace SHL a PC2 je nijak nezmění a klíče K_i si budou rovny. Tím nastane i nežádoucí rovnost $E_K = D_K$.

Slabé a poloslabé klíče

Hodně pozornosti bylo věnováno studiu klíčů. Budou-li registry C a D na počátku obsahovat konstantní bity (buď nuly nebo jedničky), pak je operace SHL a PC2 je nijak nezmění a klíče K_i si budou rovny. Tím nastane i nežádoucí rovnost $E_K = D_K$.

Celkem existují 4 tyto ***slabé klíče*** (podle toho, zda registry C nebo D obsahují nuly nebo jedničky). Podobnou vlastnost má 6 dvojic tzv. ***poloslabých klíčů*** K_1 a K_2, pro něž platí

$$E_{K_1} E_{K_2} = Id \quad \text{neboli} \quad E_{K_1} = D_{K_2}.$$

Slabé a poloslabé klíče

Hodně pozornosti bylo věnováno studiu klíčů. Budou-li registry C a D na počátku obsahovat konstantní bity (buď nuly nebo jedničky), pak je operace SHL a PC2 je nijak nezmění a klíče K_i si budou rovny. Tím nastane i nežádoucí rovnost $E_K = D_K$.

Celkem existují 4 tyto ***slabé klíče*** (podle toho, zda registry C nebo D obsahují nuly nebo jedničky). Podobnou vlastnost má 6 dvojic tzv. ***poloslabých klíčů*** K_1 a K_2, pro něž platí

$$E_{K_1} E_{K_2} = Id \quad \text{neboli} \quad E_{K_1} = D_{K_2}.$$

Ty vznikají tak, že jejich klíče jsou shodné, ale v opačném pořadí jejich použití. Tím vzniká efekt, že při šifrování druhým klíčem probíhá postupné dešifrování klíčem prvním. To nastane např. u této dvojice klíčů

$$(01FE01FE01FE01FE, FE01FE01FE01FE01).$$

Slabé a poloslabé klíče

Hodně pozornosti bylo věnováno studiu klíčů. Budou-li registry C a D na počátku obsahovat konstantní bity (buď nuly nebo jedničky), pak je operace SHL a PC2 je nijak nezmění a klíče K_i si budou rovny. Tím nastane i nežádoucí rovnost $E_K = D_K$.

Celkem existují 4 tyto ***slabé klíče*** (podle toho, zda registry C nebo D obsahují nuly nebo jedničky). Podobnou vlastnost má 6 dvojic tzv. ***poloslabých klíčů*** K_1 a K_2, pro něž platí

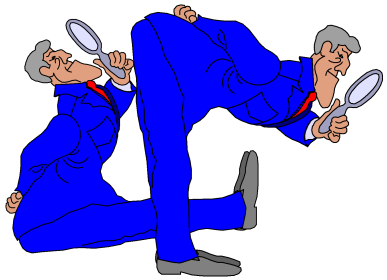
$$E_{K_1} E_{K_2} = Id \quad \text{neboli} \quad E_{K_1} = D_{K_2}.$$

Ty vznikají tak, že jejich klíče jsou shodné, ale v opačném pořadí jejich použití. Tím vzniká efekt, že při šifrování druhým klíčem probíhá postupné dešifrování klíčem prvním. To nastane např. u této dvojice klíčů

$$(01FE01FE01FE01FE, FE01FE01FE01FE01).$$

Později bylo identifikováno mnoho dalších tříd různě "***slabých***"

O čem to bude



Hellmanův časopaměťový útok I

Východisko k luštění lze nalézt i v efektivním využití času a paměti. Vychází se ze znalosti konkrétního otevřeného textu, který se velmi často objevuje ve zprávách, např. osm za sebou jdoucích mezer.

Hellmanův časopaměťový útok I

Východisko k luštění lze nalézt i v efektivním využití času a paměti. Vychází se ze znalosti konkrétního otevřeného textu, který se velmi často objevuje ve zprávách, např. osm za sebou jdoucích mezer.

Při zachycení jemu odpovídajícímu šifrového textu můžeme klíč luštit vyzkoušením všech jeho možností (čas t=256 šifrování, paměť m=1 slovo) nebo tím, že si předem připravíme tabulku všech možných dvojic (klíč, šifrový text) (čas t=1, paměť m=256 slov) a porovnáním šifrového textu. V obou případech je celková spotřeba "časopaměti" t·m=256 – spotřebu času lze přelít do paměti a naopak.

Hellmanův časopaměťový útok I

Východisko k luštění lze nalézt i v efektivním využití času a paměti. Vychází se ze znalosti konkrétního otevřeného textu, který se velmi často objevuje ve zprávách, např. osm za sebou jdoucích mezer.

Při zachycení jemu odpovídajícímu šifrového textu můžeme klíč luštit vyzkoušením všech jeho možností (čas t=256 šifrování, paměť m=1 slovo) nebo tím, že si předem připravíme tabulku všech možných dvojic (klíč, šifrový text) (čas t=1, paměť m=256 slov) a porovnáním šifrového textu. V obou případech je celková spotřeba "časopaměti" t·m=256 – spotřebu času lze přelít do paměti a naopak.

Hellman v roce 1980 přišel na to, jak část tabulky vypočítat předem a ve velmi zhuštěné podobě ji uložit do paměti. Luštění potom probíhá částečně jako šifrování a částečně jako porovnávání.

Hellmanův časopaměťový útok II

Při optimalizaci požadavků na čas, paměť a cenu dospěl k návrhu, který zahrnoval 10 000 DES čipů, paměť 1000 GB a přípravné výpočty tabulek trvající 1 až 2 roky. To vše v ceně asi 3,6 miliónu $.

Hellmanův časopaměťový útok II

Při optimalizaci požadavků na čas, paměť a cenu dospěl k návrhu, který zahrnoval 10 000 DES čipů, paměť 1000 GB a přípravné výpočty tabulek trvající 1 až 2 roky. To vše v ceně asi 3,6 miliónu $.

S těmito prostředky byl schopen určit až 1000 klíčů denně (v důsledku paralelismu) s průměrným čekáním na řešení 1 den. Při plné amortizaci zařízení do 5 let by cena za vyluštění jednoho klíče byla 1-100 $.

Hellmanův časopaměťový útok II

Při optimalizaci požadavků na čas, paměť a cenu dospěl k návrhu, který zahrnoval 10 000 DES čipů, paměť 1000 GB a přípravné výpočty tabulek trvající 1 až 2 roky. To vše v ceně asi 3,6 miliónu $.

S těmito prostředky byl schopen určit až 1000 klíčů denně (v důsledku paralelismu) s průměrným čekáním na řešení 1 den. Při plné amortizaci zařízení do 5 let by cena za vyluštění jednoho klíče byla 1-100 $.

Nevýhoda Hellmanova útoku spočívá v tom, že vše se počítá pro konkrétní otevřený text. Pro jiný otevřený text musíme všechny předběžné výpočty provést znovu.

Další zajímavé vlastnosti I - kratší perioda

Při studiu DES do r. 1990 bylo popsáno několik dalších vlastností DES. Zmiňme se o odhalení kratší periody v jednom z modů činnosti DES.

Další zajímavé vlastnosti I - kratší perioda

Při studiu DES do r. 1990 bylo popsáno několik dalších vlastností DES. Zmiňme se o odhalení kratší periody v jednom z modů činnosti DES.

Při vazbě $k = 64$ u OFB je průměrná délka cyklu produkovaného hesla přibližně 2^{63} bloků, avšak při jiné délce k je pouze kolem 2^{31} bloků!

Další zajímavé vlastnosti I - kratší perioda

Při studiu DES do r. 1990 bylo popsáno několik dalších vlastností DES. Zmiňme se o odhalení kratší periody v jednom z modů činnosti DES.

Při vazbě $k = 64$ u OFB je průměrná délka cyklu produkovaného hesla přibližně 2^{63} bloků, avšak při jiné délce k je pouze kolem 2^{31} bloků!

Toto množství hesla při rychlosti šifrování 2^{16} bitů za sekundu (např. pro digitalizovanou řeč) je vyčerpáno asi za 18 hodin!

Další zajímavé vlastnosti I - kratší perioda

Při studiu DES do r. 1990 bylo popsáno několik dalších vlastností DES. Zmiňme se o odhalení kratší periody v jednom z modů činnosti DES.

Při vazbě $k = 64$ u OFB je průměrná délka cyklu produkovaného hesla přibližně 2^{63} bloků, avšak při jiné délce k je pouze kolem 2^{31} bloků!

Toto množství hesla při rychlosti šifrování 2^{16} bitů za sekundu (např. pro digitalizovanou řeč) je vyčerpáno asi za 18 hodin!

Poté by došlo k dvojímu použití hesla, což je pro luštitele známá a jednoduchá úloha. Musí být proto používaná ***pouze plná vazba*** $k = 64$, při které je průměrná délka periody hesla dostatečná.

Další zajímavé vlastnosti II - mnohonásobné použití DES

K odstranění nedostatku malého objemu klíče bylo doporučováno ***několikanásobné šifrování DES***.

Další zajímavé vlastnosti II - mnohonásobné použití DES

K odstranění nedostatku malého objemu klíče bylo doporučováno ***několikanásobné šifrování DES***. Při dvojnásobném šifrování algoritmem DES s použitím dvou různých klíčů, tj.

$$C = E_{K_2}(E_{K_1}(M)) \tag{6.1}$$

by klíč vzrostl na 112 bitů, což se zdá být dostatečné. Avšak byla nalezena technika, která by de facto redukovala klíč na pouhých 57 bitů. Spočívá v útoku se znalostí otevřeného textu.

Další zajímavé vlastnosti II - mnohonásobné použití DES

K odstranění nedostatku malého objemu klíče bylo doporučováno ***několikanásobné šifrování DES***. Při dvojnásobném šifrování algoritmem DES s použitím dvou různých klíčů, tj.

$$C = E_{K_2}(E_{K_1}(M)) \tag{6.1}$$

by klíč vzrostl na 112 bitů, což se zdá být dostatečné. Avšak byla nalezena technika, která by de facto redukovala klíč na pouhých 57 bitů. Spočívá v útoku se znalostí otevřeného textu.

Známý otevřený text je zašifrován všemi možnými klíči (vzniká množina $E_{K_1}(M)$ o 2^{56} prvcích) a známý šifrový text je dešifrován všemi možnými klíči (vzniká množina $D_{K_2}(C)$ o 2^{56} prvcích).

Další zajímavé vlastnosti III - mnohonásobné použití DES

Průnik obou vzniklých množin obsahuje pravé řešení. Kromě toho vzniká asi 2^{48} falešných párů klíčů. Ty lze snadno vyloučit kontrolním zašifrováním na dalším známém páru (M, C). Tato metoda se nazývá "meet in the middle".

Další zajímavé vlastnosti III - mnohonásobné použití DES

Průnik obou vzniklých množin obsahuje pravé řešení. Kromě toho vzniká asi 2^{48} falešných párů klíčů. Ty lze snadno vyloučit kontrolním zašifrováním na dalším známém páru (M, C). Tato metoda se nazývá "meet in the middle".

IBM doporučení Diffieho a Hellmana přijala, vylepšila a k šifrování významných informací (jako jsou například klíče nižších úrovní) používá dvou klíčů následným způsobem

$$C = E_{K_1}(D_{K_2}(E_{K_1}(M))). \tag{6.2}$$

Další zajímavé vlastnosti III - mnohonásobné použití DES

Průnik obou vzniklých množin obsahuje pravé řešení. Kromě toho vzniká asi 2^{48} falešných párů klíčů. Ty lze snadno vyloučit kontrolním zašifrováním na dalším známém páru (M, C). Tato metoda se nazývá "meet in the middle".

IBM doporučení Diffieho a Hellmana přijala, vylepšila a k šifrování významných informací (jako jsou například klíče nižších úrovní) používá dvou klíčů následným způsobem

$$C = E_{K_1}(D_{K_2}(E_{K_1}(M))). \tag{6.2}$$

Tato metoda trojnásobného šifrování (s dvojnásobným klíčem) má výhodu v kompatibilitě se systémy s jedním klíčem (původního šifrování se dosáhne volbou $K_1 = K_2$).

Další zajímavé vlastnosti IV - DES-X

Jedním z dalších poměrně jednoduchých návrhů na zabezpečení DESu proti útoku hrubou silou je DES-X vyvinutý Rivestem.

Další zajímavé vlastnosti IV - DES-X

Jedním z dalších poměrně jednoduchých návrhů na zabezpečení DESu proti útoku hrubou silou je DES-X vyvinutý Rivestem.

Ten používá **184-*bitový klíč***, který se skládá z jednoho 56-bitového DES klíče K a dva 64-bitové klíčů K_1 a K_2.

Další zajímavé vlastnosti IV - DES-X

Jedním z dalších poměrně jednoduchých návrhů na zabezpečení DESu proti útoku hrubou silou je DES-X vyvinutý Rivestem.

Ten používá **184-*bitový klíč***, který se skládá z jednoho 56-bitového DES klíče K a dva 64-bitové klíčů K_1 a K_2.

Šifrování se provede nejprve pouhým XORováním K_1 se zprávou, pak šifrováním s DESem pomocí klíče K a konečně XORováním s K_2, tedy kryptogram pro 64-bitový blok zprávy M je

$$C = K_2 \oplus DES_K(M \oplus K_1),$$

kde $DES_K(-)$ je DES šifrování s klíčem K.

Další zajímavé vlastnosti V - DES-X

Tento systém je srovnatelný s DESem z hlediska účinnosti a je také zpětně kompatibilní s DES, prostě předpokládejme, že K_1 a K_2 jsou nulové vektory. Bylo prokázáno, že DES-X je v podstatě imunní vůči hrubou silou klíčových vyhledávání.

Další zajímavé vlastnosti V - DES-X

Tento systém je srovnatelný s DESem z hlediska účinnosti a je také zpětně kompatibilní s DES, prostě předpokládejme, že K_1 a K_2 jsou nulové vektory. Bylo prokázáno, že DES-X je v podstatě imunní vůči hrubou silou klíčových vyhledávání.

Ačkoli DES-X neposkytuje zvýšenou bezpečnost proti teoretickým diferenciálním a lineárním útokům, pokud si myslíte, že jediný způsob, jak zlomit DES, je hrubou silou, pak DES-X je atraktivní nahrazení DESu.

Analytické útoky I - diferenciální kryptoanalýza

Obránci DES, kteří argumentovali tím, že doposud nebyl předložen žádný vážnější útok proti DES (a přehlíželi tak výše uvedené skutečnosti), mají od 11.8. 1990 ztíženou argumentaci.

Analytické útoky I - diferenciální kryptoanalýza

Obránci DES, kteří argumentovali tím, že doposud nebyl předložen žádný vážnější útok proti DES (a přehlíželi tak výše uvedené skutečnosti), mají od 11.8. 1990 ztíženou argumentaci.

Tento den předložili Eli Biham a Adi Shamir z Weizmanova institutu v Izraeli metodu luštění, nazývanou diferenciální kryptoanalýza (DK). Je analytickým útokem proti DES.

Analytické útoky I - diferenciální kryptoanalýza

Obránci DES, kteří argumentovali tím, že doposud nebyl předložen žádný vážnější útok proti DES (a přehlíželi tak výše uvedené skutečnosti), mají od 11.8. 1990 ztíženou argumentaci.

Tento den předložili Eli Biham a Adi Shamir z Weizmanova institutu v Izraeli metodu luštění, nazývanou diferenciální kryptoanalýza (DK). Je analytickým útokem proti DES.

Snadno s ní **rozbili osmikrokovou DES, japonskou blokovou šifru FEAL a předchůdce DES LUCIFERa**. U patnáctikrokové verze DES s ní dosáhli lepších výsledků, než je zkoušení všech možných klíčů.

Analytické útoky I - diferenciální kryptoanalýza

Obránci DES, kteří argumentovali tím, že doposud nebyl předložen žádný vážnější útok proti DES (a přehlíželi tak výše uvedené skutečnosti), mají od 11.8. 1990 ztíženou argumentaci.

Tento den předložili Eli Biham a Adi Shamir z Weizmanova institutu v Izraeli metodu luštění, nazývanou diferenciální kryptoanalýza (DK). Je analytickým útokem proti DES.

Snadno s ní **rozbili osmikrokovou DES, japonskou blokovou šifru FEAL a předchůdce DES LUCIFERa**. U patnáctikrokové verze DES s ní dosáhli lepších výsledků, než je zkoušení všech možných klíčů.

Po dvou letech pak metodu zdokonalili i pro plně šestnáctikrokovou verzi DES a tím vyvrátili tvrzení Davise.

Analytické útoky II - diferenciální kryptoanalýza

Podstata diferenciální kryptografie spočívá ve využití nevhodných S-boxů a slabého zpracování klíče. Vliv klíče K_i se dá totiž určitým způsobem "***vyblokovat***".

Analytické útoky II - diferenciální kryptoanalýza

Podstata diferenciální kryptografie spočívá ve využití nevhodných S-boxů a slabého zpracování klíče. Vliv klíče K_i se dá totiž určitým způsobem "***vyblokovat***".

Uvažujme v jednom kroku DES při výpočtu $f(R, K)$ zpracování dvou vstupů: R_1 a R_2. První úvaha spočívá v tom, že diference R_1 a R_2 (=$R_1 \oplus R_2$) zůstává nezměněna, když R_1 a R_2 projdou rozšířením (expanzí) E a operací $\oplus$ s klíčem K.

Analytické útoky II - diferenciální kryptoanalýza

Podstata diferenciální kryptografie spočívá ve využití nevhodných S-boxů a slabého zpracování klíče. Vliv klíče K_i se dá totiž určitým způsobem "***vyblokovat***".

Uvažujme v jednom kroku DES při výpočtu $f(R, K)$ zpracování dvou vstupů: R_1 a R_2. První úvaha spočívá v tom, že diference R_1 a R_2 (=$R_1 \oplus R_2$) zůstává nezměněna, když R_1 a R_2 projdou rozšířením (expanzí) E a operací $\oplus$ s klíčem K.

Vliv klíče se v této diferenci eliminuje:

$$(E(R_1) \oplus K) \oplus (E(R_2) \oplus K) = E(R_1) \oplus E(R_2) = E(R_1 \oplus R_2)! \quad (6.3)$$

Analytické útoky III - S-boxy

Druhá myšlenka spočívá v tom, jak obejít nelineární S-boxy.

Analytické útoky III - S-boxy

Druhá myšlenka spočívá v tom, jak obejít nelineární S-boxy.

Pro některé diference vstupů jsou mnohé diference výstupů z S-boxů málo pravděpodobné a jiné velmi pravděpodobné.

Analytické útoky III - S-boxy

Druhá myšlenka spočívá v tom, jak obejít nelineární S-boxy.

Pro některé diference vstupů jsou mnohé diference výstupů z S-boxů málo pravděpodobné a jiné velmi pravděpodobné.

Výstupní diference se ovšem v dalším kroku schématu stávají vstupními diferencemi! Permutace P nám v tom vůbec nevadí. Tímto postupným zřetězením vstupně-výstupních diferencí dostáváme nakonec pravděpodobnostní vztah mezi diferencemi otevřeného textu a diferencemi šifrového textu.

Analytické útoky III - S-boxy

Druhá myšlenka spočívá v tom, jak obejít nelineární S-boxy.

Pro některé diference vstupů jsou mnohé diference výstupů z S-boxů málo pravděpodobné a jiné velmi pravděpodobné.

Výstupní diference se ovšem v dalším kroku schématu stávají vstupními diferencemi! Permutace P nám v tom vůbec nevadí. Tímto postupným zřetězením vstupně-výstupních diferencí dostáváme nakonec pravděpodobnostní vztah mezi diferencemi otevřeného textu a diferencemi šifrového textu.

Nalezneme-li páry (M, C), kde M je otevřený text, C je šifrový text, které vyhovují našemu modelu, máme o průběhu šifrování už dost informací.

Analytické útoky III - S-boxy

Druhá myšlenka spočívá v tom, jak obejít nelineární S-boxy.

Pro některé diference vstupů jsou mnohé diference výstupů z S-boxů málo pravděpodobné a jiné velmi pravděpodobné.

Výstupní diference se ovšem v dalším kroku schématu stávají vstupními diferencemi! Permutace P nám v tom vůbec nevadí. Tímto postupným zřetězením vstupně-výstupních diferencí dostáváme nakonec pravděpodobnostní vztah mezi diferencemi otevřeného textu a diferencemi šifrového textu.

Nalezneme-li páry (M, C), kde M je otevřený text, C je šifrový text, které vyhovují našemu modelu, máme o průběhu šifrování už dost informací.

Určení klíčů K_{16} až K_1 je pak jen složitou technickou záležitostí.

Analytické útoky IV - S-boxy

S každým krokem se ovšem příslušné pravděpodobnosti násobí, takže čím více kroků má schéma, tím je účinnost metody horší.

Analytické útoky IV - S-boxy

S každým krokem se ovšem příslušné pravděpodobnosti násobí, takže čím více kroků má schéma, tím je účinnost metody horší.

Například na **rozbití osmikrokové verze DES stačily pouze 2 minuty**, na šestnáctikrokovou verzi je to již 2^{37} operací.

Analytické útoky IV - S-boxy

S každým krokem se ovšem příslušné pravděpodobnosti násobí, takže čím více kroků má schéma, tím je účinnost metody horší.

Například na **rozbití osmikrokové verze DES stačily pouze 2 minuty**, na šestnáctikrokovou verzi je to již 2^{37} operací.

Dosud je také k tomu potřeba velké množství odpovídajících si dvojic (M, C). Na druhé straně se metoda neustále zdokonaluje, takže nelze odhadnout, kde se zastaví její účinnost.

Analytické útoky IV - S-boxy

S každým krokem se ovšem příslušné pravděpodobnosti násobí, takže čím více kroků má schéma, tím je účinnost metody horší.

Například na **rozbití osmikrokové verze DES stačily pouze 2 minuty**, na šestnáctikrokovou verzi je to již 2^{37} operací.

Dosud je také k tomu potřeba velké množství odpovídajících si dvojic (M, C). Na druhé straně se metoda neustále zdokonaluje, takže nelze odhadnout, kde se zastaví její účinnost.

Na obranu proti ní v dnešní podobě zatím postačuje trojnásobné šifrování. Avšak důvěra v DES jako celek dostala vážnou trhlinu.

Analytické útoky V

Ve dnech 23.-27.května 1993 se v Lofthusu, malé norské vesničce, konala další ze série konferencí Mezinárodní asociace pro kryptologický výzkum - EUROCRYPT'93. K DES se vztahovalo několik příspěvků. Uveďme dva nejdůležitější.

Analytické útoky V

Ve dnech 23.-27.května 1993 se v Lofthusu, malé norské vesničce, konala další ze série konferencí Mezinárodní asociace pro kryptologický výzkum - EUROCRYPT'93. K DES se vztahovalo několik příspěvků. Uveďme dva nejdůležitější.

První z příspěvků přednesl Eli Biham, jeden ze známých izraelských "rozbíječů" DES. Svůj příspěvek s názvem "Nové typy kryptoanalytických útoků na bázi příbuzných klíčů" uvedl předvedením čersvého výtisku knihy o diferenciální kryptoanalýze DES; napsal ji společně s Adi Shamirem.

Analytické útoky V

Ve dnech 23.-27.května 1993 se v Lofthusu, malé norské vesničce, konala další ze série konferencí Mezinárodní asociace pro kryptologický výzkum - EUROCRYPT'93. K DES se vztahovalo několik příspěvků. Uveďme dva nejdůležitější.

První z příspěvků přednesl Eli Biham, jeden ze známých izraelských "rozbíječů" DES. Svůj příspěvek s názvem "Nové typy kryptoanalytických útoků na bázi příbuzných klíčů" uvedl předvedením čersvého výtisku knihy o diferenciální kryptoanalýze DES; napsal ji společně s Adi Shamirem.

Biham si všiml, že některé rodiny kryptoschémat využívají poměrně jednoduché tvorby rundovských klíčů (u DES viz K_i), čímž mezi nimi vznikají zřejmé vztahy, a ty pak využil ke dvěma útokům.

Analytické útoky VI

Prvním útokem "***s možností volby otevřeného textu***" dosáhl novou redukci složitosti při luštění poměrně široké třídy kryptoschémat a předvedl rychlejší variantu tohoto útoku s využitím ***vlastnosti komplementárnosti***.

Analytické útoky VI

Prvním útokem "***s možností volby otevřeného textu***" dosáhl novou redukci složitosti při luštění poměrně široké třídy kryptoschémat a předvedl rychlejší variantu tohoto útoku s využitím ***vlastnosti komplementárnosti***.

Druhý útok je nový a autor ho pojmenoval "***útok s možností volby vztahů v klíči s nízkou složitostí***".

Analytické útoky VII

Připomeňme rozlišení typů útoků na šifrovací systém

- ***Luštění pouze ze šifrového textu***. Kryptoanalytik má k dispozici libovolné množství kryptogramů, neví ovšem, jaký otevřený text byl zašifrován.

Analytické útoky VII

Připomeňme rozlišení typů útoků na šifrovací systém

- ***Luštění pouze ze šifrového textu***. Kryptoanalytik má k dispozici libovolné množství kryptogramů, neví ovšem, jaký otevřený text byl zašifrován.
- ***Luštění při znalosti otevřeného a šifrového textu.*** Kryptoanalytik má k dispozici libovolné množství dvojic otevřeného textu a jemu příslušného kryptogramu.

Analytické útoky VII

Připomeňme rozlišení typů útoků na šifrovací systém

- ***Luštění pouze ze šifrového textu***. Kryptoanalytik má k dispozici libovolné množství kryptogramů, neví ovšem, jaký otevřený text byl zašifrován.
- ***Luštění při znalosti otevřeného a šifrového textu.*** Kryptoanalytik má k dispozici libovolné množství dvojic otevřeného textu a jemu příslušného kryptogramu.
- ***Luštění s možností volby.***

Analytické útoky VII

Připomeňme rozlišení typů útoků na šifrovací systém

- ***Luštění pouze ze šifrového textu***. Kryptoanalytik má k dispozici libovolné množství kryptogramů, neví ovšem, jaký otevřený text byl zašifrován.
- ***Luštění při znalosti otevřeného a šifrového textu.*** Kryptoanalytik má k dispozici libovolné množství dvojic otevřeného textu a jemu příslušného kryptogramu.
- ***Luštění s možností volby.***Kryptoanalytik má možnost vnutit šifrovacímu systému svůj otevřený text a obdržet od něj odpovídající kryptogram.

Analytické útoky VII

Připomeňme rozlišení typů útoků na šifrovací systém

- ***Luštění pouze ze šifrového textu***. Kryptoanalytik má k dispozici libovolné množství kryptogramů, neví ovšem, jaký otevřený text byl zašifrován.
- ***Luštění při znalosti otevřeného a šifrového textu.*** Kryptoanalytik má k dispozici libovolné množství dvojic otevřeného textu a jemu příslušného kryptogramu.
- ***Luštění s možností volby.***Kryptoanalytik má možnost vnutit šifrovacímu systému svůj otevřený text a obdržet od něj odpovídající kryptogram.

Analytické útoky VIII

Jaký útok kryptoanalytik zvolí, záleží vždy na konkrétní úloze, před kterou je postaven. V tomto typu útoku bylo využito toho, že určité jednoduché vztahy mezi klíči mají za následek určité jednoduché vztahy mezi otevřenými a šifrovými texty.

Analytické útoky VIII

Jaký útok kryptoanalytik zvolí, záleží vždy na konkrétní úloze, před kterou je postaven. V tomto typu útoku bylo využito toho, že určité jednoduché vztahy mezi klíči mají za následek určité jednoduché vztahy mezi otevřenými a šifrovými texty.

Bihamovy útoky jsou aplikovatelné na rodinu schémat LOKI a LUCIFERA. Mohly by být aplikovatelné i na DES, kdyby posuny registrů C a D v algoritmu přípravy klíčů K_i byly stejné ....

Analytické útoky VIII

Jaký útok kryptoanalytik zvolí, záleží vždy na konkrétní úloze, před kterou je postaven. V tomto typu útoku bylo využito toho, že určité jednoduché vztahy mezi klíči mají za následek určité jednoduché vztahy mezi otevřenými a šifrovými texty.

Bihamovy útoky jsou aplikovatelné na rodinu schémat LOKI a LUCIFERA. Mohly by být aplikovatelné i na DES, kdyby posuny registrů C a D v algoritmu přípravy klíčů K_i byly stejné ....

Jak křehká je bezpečnost DES! Ukazuje se, že **nové výsledky útoků** nepadají z nebe, ale **jsou pouze výsledkem nového soustředěného úsilí** kryptoanalytiků.

Analytické útoky VIII

Jaký útok kryptoanalytik zvolí, záleží vždy na konkrétní úloze, před kterou je postaven. V tomto typu útoku bylo využito toho, že určité jednoduché vztahy mezi klíči mají za následek určité jednoduché vztahy mezi otevřenými a šifrovými texty.

Bihamovy útoky jsou aplikovatelné na rodinu schémat LOKI a LUCIFERA. Mohly by být aplikovatelné i na DES, kdyby posuny registrů C a D v algoritmu přípravy klíčů K_i byly stejné ....

Jak křehká je bezpečnost DES! Ukazuje se, že **nové výsledky útoků** nepadají z nebe, ale **jsou pouze výsledkem nového soustředěného úsilí** kryptoanalytiků.

Bihamův útok je totiž zlepšením, rozšířením a zobecněním Knudsenova útoku na australský algoritmus LOKI91 z roku 1992, ale vznikl nezávisle na něm.

Analytické útoky IX - lineární kryptoanalýza

Druhý příspěvek byl ještě zajímavější. Přednesl ho Japonec **M. Matsui** a byl nazván "*Lineární kryptoanalytická metoda pro šifru DES*". Nová metoda kryptoanalýzy je útokem "se znalostí otevřeného textu".

Analytické útoky IX - lineární kryptoanalýza

Druhý příspěvek byl ještě zajímavější. Přednesl ho Japonec **M. Matsui** a byl nazván "*Lineární kryptoanalytická metoda pro šifru DES*". Nová metoda kryptoanalýzy je útokem "se znalostí otevřeného textu".

Její podstata spočívá v ***objevu lineárních vztahů v kryptosystému DES (s využitelnou pravděpodobností)***. To se mu podařilo díky ***slabým nelinearitám*** S-boxů.

Analytické útoky IX - lineární kryptoanalýza

Druhý příspěvek byl ještě zajímavější. Přednesl ho Japonec **M. Matsui** a byl nazván "*Lineární kryptoanalytická metoda pro šifru DES*". Nová metoda kryptoanalýzy je útokem "se znalostí otevřeného textu".

Její podstata spočívá v ***objevu lineárních vztahů v kryptosystému DES (s využitelnou pravděpodobností)***. To se mu podařilo díky ***slabým nelinearitám*** S-boxů.

Následující výsledky kryptoanalýzy byly získány na pracovní stanici Hewlett-Packard HP 9750 (PA-RISC/66 MHz) s programy vytvořenými v jazyku C.

Analytické útoky X - lineární kryptoanalýza

1. Výsledky experimentů při útoku se znalostí otevřeného textu:
   - osmikroková DES je rozluštitelná s 2^{21} otevřenými texty za 40 sekund,

Analytické útoky X - lineární kryptoanalýza

1. Výsledky experimentů při útoku se znalostí otevřeného textu:
   - osmikroková DES je rozluštitelná s 2^{21} otevřenými texty za 40 sekund,
   - dvanáctikroková DES je rozluštitelná s 2^{33} otevřenými texty za 50 hodin,

Analytické útoky X - lineární kryptoanalýza

1. Výsledky experimentů při útoku se znalostí otevřeného textu:
   - osmikroková DES je rozluštitelná s 2^{21} otevřenými texty za 40 sekund,
   - dvanáctikroková DES je rozluštitelná s 2^{33} otevřenými texty za 50 hodin,
   - šestnáctikroková DES je rozluštitelná s 2^{47} známými otevřenými texty rychleji, než je vyčepávající hledání 56-bitového klíče.

Analytické útoky X - lineární kryptoanalýza

1. Výsledky experimentů při útoku se znalostí otevřeného textu:
   - osmikroková DES je rozluštitelná s 2^{21} otevřenými texty za 40 sekund,
   - dvanáctikroková DES je rozluštitelná s 2^{33} otevřenými texty za 50 hodin,
   - šestnáctikroková DES je rozluštitelná s 2^{47} známými otevřenými texty rychleji, než je vyčepávající hledání 56-bitového klíče.

V některých případech (není-li otevřený text náhodný) lze klíč luštit přímo ze šifrového textu. To je novinka, která přímo ohromuje. Přitom Matsuiho metoda luštění je myšlenkovitě přímočará.

Analytické útoky X - lineární kryptoanalýza

2. Zde jsou výsledky experimentů pro luštění pouze ze šifrového textu:

Analytické útoky X - lineární kryptoanalýza

2. Zde jsou výsledky experimentů pro luštění pouze ze šifrového textu:
   - je-li otevřený text tvořen anglickými texty ve znacích ASCII, osmikroková DES je rozluštitelná s 2^{29} šifrovými texty,

Analytické útoky X - lineární kryptoanalýza

2. Zde jsou výsledky experimentů pro luštění pouze ze šifrového textu:
   - je-li otevřený text tvořen anglickými texty ve znacích ASCII, osmikroková DES je rozluštitelná s 2^{29} šifrovými texty,
   - jsou-li otevřené texty náhodné znaky ASCII, je potřeba 2^{37} šifrových textů,

Analytické útoky X - lineární kryptoanalýza

2. Zde jsou výsledky experimentů pro luštění pouze ze šifrového textu:
   - je-li otevřený text tvořen anglickými texty ve znacích ASCII, osmikroková DES je rozluštitelná s 2^{29} šifrovými texty,
   - jsou-li otevřené texty náhodné znaky ASCII, je potřeba 2^{37} šifrových textů,
   - jsou-li otevřené texty tvořeny zcela náhodnými znaky, existují situace, kdy je Matsuiho metoda rychlejší než vyzkoušení všech hodnot klíče.

Analytické útoky X - lineární kryptoanalýza

2. Zde jsou výsledky experimentů pro luštění pouze ze šifrového textu:
   - je-li otevřený text tvořen anglickými texty ve znacích ASCII, osmikroková DES je rozluštitelná s 2^{29} šifrovými texty,
   - jsou-li otevřené texty náhodné znaky ASCII, je potřeba 2^{37} šifrových textů,
   - jsou-li otevřené texty tvořeny zcela náhodnými znaky, existují situace, kdy je Matsuiho metoda rychlejší než vyzkoušení všech hodnot klíče.

Z příspěvku je zřejmé, že poslední slovo ve využití této myšlenky ještě nepadlo. Dává zcela konkrétní metodu pro luštění algoritmu DES, ale má zejména význam vědecko-metodologický. Příspěvek obsahuje mnoho nových myšlenek, které budou studovány a použity jak pro kryptoanalýzu, tak pro tvorbu nových kryptoschémat.