Тема 1. Вступ. Основні поняття та положення комп'ютерної криптографії.

Для позначення всієї оласті таємного зв'язку використовується термін криптологія, який походить від грецьких коренів «cryptos» – таємний та «logos» – повідомлення. Криптологія поділяється на дві області: криптографію та криптоаналіз. Завдання криптографа – забезпечити конфіденційність та аутентичність повідомлень, які передаються по каналах зв'язку. Завдання криптоаналітика – зламати систему захисту, розроблену криптографами без знання ключа. Ключ – це певний секретний стан деяких параметрів алгоритму криптографічного перетворення даних, які забезпечують вибір тільки одного варіанту із всіх можливих варіантів для даного алгоритму. На даний час розрізняється два класи криптосистем: симетричні одноключові криптосистеми (з секретним ключем) та асиметричні двохключові криптосистеми (з відкритим ключем). В симетричних криптосистемах один і той самий ключ використовується як для шифрування, так і для розшифровування даних. В асиметричних криптосистемах для шифрування і розшифровування використовуються різні, але взаємопов'язані, ключі, причому визначити один ключ, знаючи інший, практично неможливо.

До шифрів, які використовуються для криптографічного захисту інформації, представляється ряд певних вимог:

1. Достатня криптостійкість.
2. Простота шифрування та розшифровування.
3. Незначна надлишковість інформації в зв'язку з шифруванням.
4. Нечуттєвість до незначних помилок шифрування.

Криптологія є частиною такої науки, яка називається захистом інформації, яка охоплює, крім теоретичних основ, також технічні засоби, юридичні аспекти тощо. Тайнопис також є ширшим поняттям, оскільки охоплює також приховування самого факту існування повідомлень.

Будь–яка спроба зі сторони зловмисника розшифрувати шифртекст для отримання відкритого тексту або зашифрувати свій власний текст для отримання правдоподібного шифртексту, не знаючи істинного ключа, називається криптоаналітичною атакою. Фундаментальне правило криптоаналізу, вперше сформульоване у 19 ст. голандцем А.Керкхоффом полягає в тому, стійкість шифру або криптосистеми повинна визначатися тільки секретністю ключа. Іншими словами, це означає, що весь алгоритм шифрування, крім секретного ключа, відомий криптоаналітику зловмисника. Це пояснюється тим, що криптосистема, яка являє собою сукупність апаратних і програмних засобів, яку можна змінити тільки при значних затратах часу і засобів, тоді як ключ змінюється дуже легко.

Існують такі типи криптоаналітичних атак:

1. Криптоаналітична атака при наявності тільки відомого шифртексту.
2. Криптоаналітична атака при наявності відомого відкритого тексту.
3. Криптоаналітична атака при можливості вибору відкритого тексту.
4. Криптоаналітична атака з адаптивним вибором відкритого тексту.
5. Криптоаналітична атака з використанням вибраного шифртексту.
6. Криптоаналітична атака методом повного перебору всіх можливих ключів (брутальна атака).

Тема 2. Шифри перестановки та простої заміни.

При шифруванні перестановкою символи відкритого тексту переставляються за визначеним правилом в межах блоку цього тексту. Шифри перестановки є найпростішими та найдревнішими шифрами.

Найдревніший шифр – шифр скитала використовувався в 5 ст. до нашої ери правителями Спарти. На циліндричний стержень спіраллю намотувалась стрічка пергаменту і вздовж стержня писали повідомлення. Тді пергамент знімали і отримували хаотично розміщені букви. Ключем був діаметр валика.

Пізніше почали використовувати шифр частоколу. Наприклад:

р п о р ф я
к и т г а і

Отримується шифртекст: рпосфякитгаі. Ключем є висота частоколу. Зокрема, для частоколу висотою 3 маємо (ліворуч):

и г і и о а я
р п о р ф я р т р і
к т а к п г ф

Отримується шифртекст: игірпорфякта. Це складний частокіл. При використанні простого частоколу отримуємо: иоаяртрікпгф. аналогічно можна використати частокіл і більшої висоти.

З кінця 14 ст. виникли шифруючі таблиці. Наприклад, відкритий текст записується в таблицю по стовбцях, а читається по рядках. Ключем є розмір таблиці. Їх удосконаленням стали шифруючі таблиці з ключовими словами, коли стовбці та рядки переставляються у відповідності до цих ключових слів.

		л	і	т	о
		2	1	4	3
з	2	п	р	и	л
и	3	і	т	а	ю
м	4	в	о	с	ь
а	1	м	о	г	о

		і	л	о	т
		1	2	3	4
з	2	р	п	л	и
и	3	т	і	ю	а
м	4	о	в	ь	с
а	1	о	м	о	г

1	о	м	о	г
2	р	п	л	и
3	т	і	ю	а
4	о	в	ь	с

В середні віки використовувалося також шифрування за допомогою магічних квадратів. Це квадратні таблиці з вписаними в клітинки послідовностями натуральних чисел, починаючи з 1, щоб їх сума по стовбцях, рядках та діагоналях дорівнювала одному і тому самому числу.

16	3	2	13
5	10	11	8
9	6	7	12
4	15	14	1

о	и	р	м
і	о	с	ю
в	т	а	ь
л	г	о	п

Якщо не враховувати повороти, то існує тільки один квадрат розміром 3х3, 880 – розміром 4х4, біля 250000 – розміром 5х5.

Шифр Кардано являє квадрат з клітинками, частина яких вирізані. Цей квадрат накладають на такий самий суцільний квадрат і у вирізані клітинки вписують текст. Потім верхній квадрат повертають на 90 градусів і відкриваються нові клітинки. Таким чином заповнюється вся таблиця. Ключем у шифрі Кардано є розміщення вирізаних клітин. У кожному рядку і стовбці може бути вирізана тільки одна клітинка і кількість клітинок у рядку і стовбці має бути парною.

До шифрів простої заміни відноситься шифр Цезаря, шифр Цезаря з ключовим словом, полібіанський квадрат, шифруючі таблиці Трисемуса.

Тема 3. Шифри складної заміни. Шифр одноразового блокноту.

Шифри складної заміни називають багатоалфавітними, оскільки для шифрування кожного символа вихідного повідомлення використовують свій шифр простої заміни.

Шифр Гронсфельда являє собою модифікацію шифра Цезаря з числовим ключем. Під буквами вихідного повідомлення записують цифри ключового слова. Якщо ключ коротший, то його запис циклічно повторюють. Шифртекст отримують аналогічно шифру зсуву, але кожен символ зсувають на ту кількість знаків, яка записана під символом. Потрібно відмітити, що шифр Гронсфельда зламується досить легко, але його можна вдосконалити, зокрема, подвійним шифруванням різними ключами.

Біграмний шифр Плейфейра, винайдений у 1854 році, використовує прямокутну таблицю з хаотично або з ключовим словом вписаними буквами алфавіту. Відкритий текст розбивається на біграми. Він повинен мати парну кількість символів і не містити біграм з однаковими буквами. В таблиці шукаються букви біграми і вважається, що вони є вершинами прямокутника. У двох інших вершинах будуть лежати дві букви шифртексту. Якщо букви відкритого тексту потрапляють в один рядок чи стовбець, то вибираються букви, що лежать під ними, або, відповідно, ліворуч.

Для усунення такого недоліку використовується подвійний квадрат Уїтстона, в якому використовується дві прямокутних таблиці з розміщеними буквами алфавіту. Букви відкритого тексту шукаються в різних таблицях і аналогічно утворюються прямокутники. Тепер букви відкритого тексту не потраплять в один стовбець, але можуть потрапити в один рядок. Для усунення цього недоліку використовується шифр чотирьох квадратів, розміщених в квадратах. Букви відкритого тексту шукаються в діагонально протилежних квадратах, в інших квадратах шукаються букви шифртексту. Тепер ні в один рядок, ні в один стовбець букви відкритого тексту не потраплять.

Узагальненням цих шифрів є шифр Віженера. він представляється квадратною таблицею Віженера, розмірність якої відповідає кількості букв в алфавіті. По горизонталі розміщуються букви відкритого тексту, по вертикалі – букви ключа. На їх перетині отримуємо букви шифртексту. Його удосконаленням є шифр Віженера з автоключем.

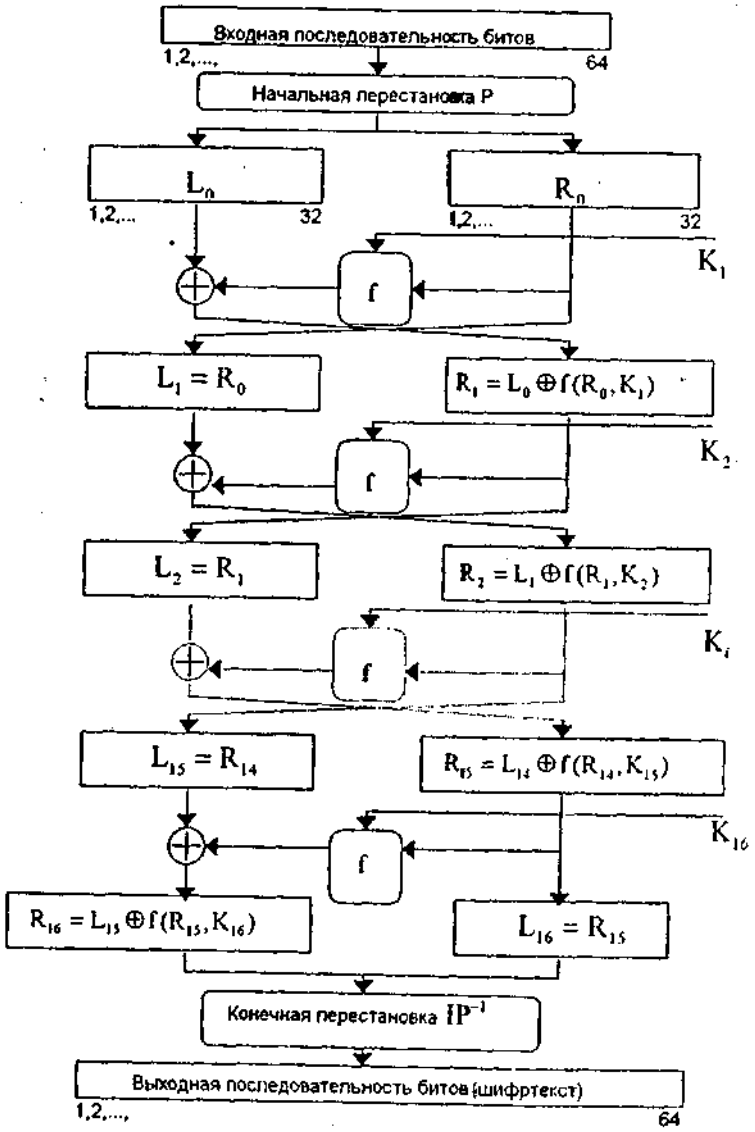
Для застосування шифру одноразового блокноту відкритий текст та ключ переводять у цифрову форму. Кожній букві відповідає її номер в алфавіті, нумерація починається з нуля. Потім це число переводять у двійкову форму. Для шифрування використовується додавання бітів по модулю 2. Операція позначається $\oplus$ і задається так: $0\oplus0=0$, $1\oplus0=1$, $0\oplus1=1$, $1\oplus1=1$. Ключем може служити довільне двійкове слово однакової довжини з відкритим текстом. Криптотекст отримують побітовим додаванням відкритого тексту та ключа за модулем 2. Дешифрування збігається із шифруванням. Щоб отримати відкритий текст, до криптотексту знову потрібно додати двійковий ключ. Шифр

одноразового блокноту не є однозначним, оскільки той самий шифртекст можна отримати для деякого іншого відкритого тексту та іншого ключа.

Назва шифру походить від того, що агент, який здійснював шифрування вручну, отримував свої копії ключів, записаними в блокнот. Якщо ключ застосовувався, то сторінка з ним знищувалась.

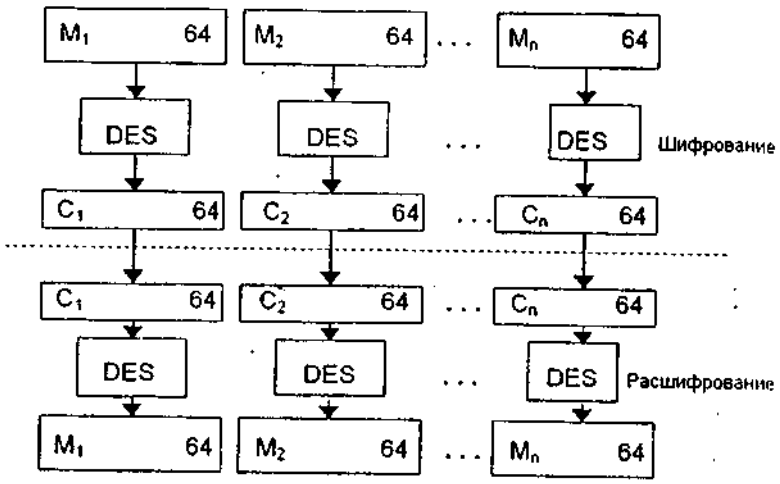
Тема 4. Алгоритм DES. Режими його роботи.

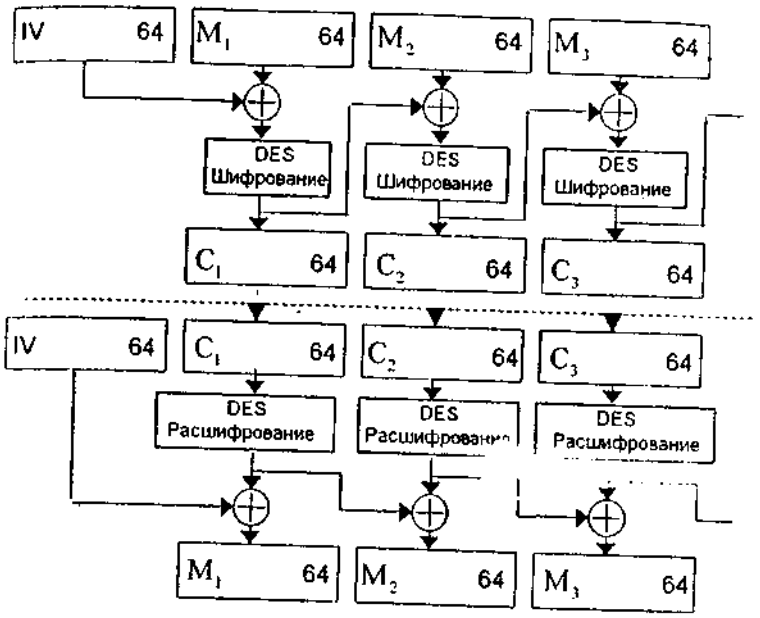
Стандарт шифрування даних DES (Data Encryption Standard) опублікований у 1977 році Національним бюро стандартів США і призначений для захисту від несанкціонованого доступу до важливої, але несекретної інформації в державних і комерційних установах США. При його використано такі позначення: L і R – ліва та права послідовності бітів; LR – конкатенація послідовностей; ⊕ – побітове додавання по модулю 2. Алгоритм використовує комбінацію підстановок і перестановок. Шифрування здійснюється 64–бітовими блоками за допомогою 64–бітового ключа, в якому значущими є 56 біт, решта 8 – перевірочні. На рис. зображено структуру алгоритму DES.

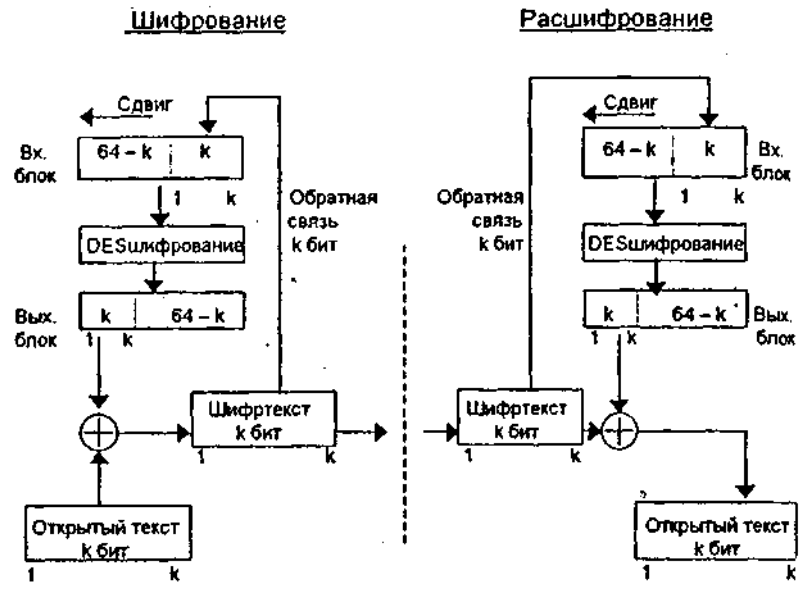


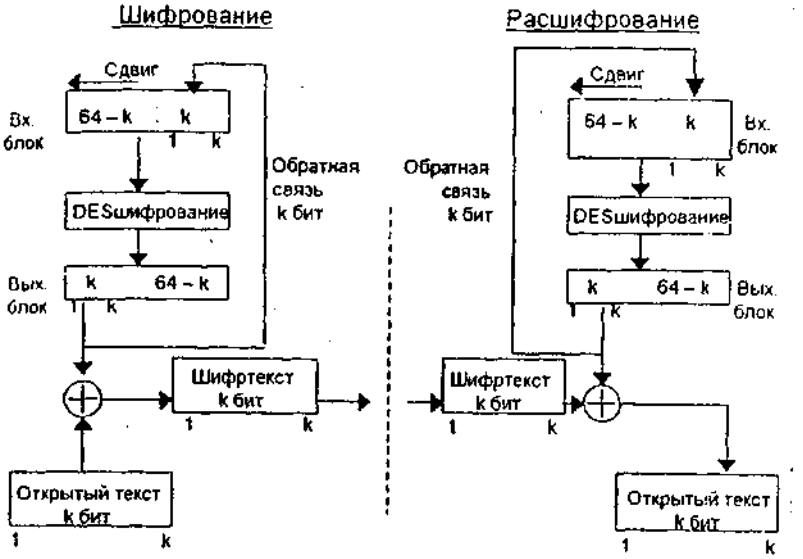
Початковий 64–бітовий блок перетворюється за допомогою матриці початкової перестановки. Потім виконується ітеративний процес шифрування, який складається з 16 циклів. Після них виконується кінцева перестановки у відповідності з матрицею кінцевої перестановки. Для виконання алгоритму потрібно 16 ключів, які генеруються з 56–бітового ключа.

Існує 4 режими роботи алгоритму DES: електронна кодова книга (ECB), зчеплення блоків шифру (CBC), зворотній зв'язок по шифртексту (CFB), зворотній зв'язок по виходу (OFB). Їх схеми відповідно зображені на рисунках.









Тема 5. Алгоритм IDEA та вітчизняний стандарт шифрування ГОСТ28147–89.

Алгоритм IDEA (International Data Encryption Algorithm) є блоковим шифром. Він оперує 64–бітовими блоками відкритого тексту. Його ключ має 128 біт. Один і той же алгоритм використовується і для шифрування, і для розшифровування. В алгоритмі використовуються такі математичні операції: побітове додавання по модулю 2; додавання беззнакових цілих по модулю 2^{16}; множення цілих по модулю $(2^{16}+1)$. Всі операції виконуються над 16–бітовими підблоками. Ці три операції несумісні в тому, що ніяка пара з цих трьох операцій не задовольняє асоціативному та дистрибутивному законам. Всього виконується 8 циклів.

Комбінування цих трьох операцій забезпечує комплексне перетворення входу, істотно затрудняючи криптоаналіз IDEA в порівнянні з DES, який базується тільки на побітовому додаванні по модулю 2.

Алгоритм IDEA використовує 52 підключі (по шість для кожного з 8 циклів і чотири для перетворення виходу). Розшифрування здійскнюється в зворотньому порядку.

Алгоритм IDEA може працювати в тих же режимах, що і DES, однак має ряд переваг. Він значно безпечніший DES, оскільки має вдвічі більший ключ. Його внутрішня структура забезпечує кращу стійкість до криптоаналізу. Програмні реалізації IDEA приблизно вдвічі швидші, ніж DES.

Стандарт шифрування ГОСТ 28147–89 являє собою 64–бітовий блочний алгоритм з 256–бітовим ключем. Використовуються такі операції: побітове додавання по модулю 2; операція додавання по модулю 2^{32} двох 32–розрядних двійкових чисел; операція додавання двох 32–розрядних чисел за модулем 2^{32}-1.

Алгоритм передбачає чотири режими роботи:

1. Шифрування даних в режимі простої заміни.
2. Шифрування даних в режимі гамування.
3. Шифрування даних в режимі гамування із зворотним зв’язком.
4. Вироблення імітовставки.

Тема 6. Алгоритми RC 2, RC 4, RC 5, RC 6.

RC–4 являє собою потоковий шифр із змінною довжиною ключа. Алгоритм працює в режимі зворотнього зв’язку по виходу (OFB). Ключова послідовність не залежить від вихідного тексту. Структура алгоритму включає блок заміни розмірністю 8x8, який являє собою залежну від ключа перестановку чисел 0, ..., 255 змінної довжини. Є два лічильники i та j, початкове значення яких дорівнює 0. На початку генерується псевдовипадковий байт, який потім додається по модулю 2 з байтом вихідного тексту для отримання шифртексту. Ініціалізація блоку заміни виконується за допомогою ключа. За програмною реалізацією алгоритм RC–4 приблизно в 10 разів швидший від DES. Можливі узагальнення алгоритму на більшу довжину слів і розмір блоку заміни. Можна побудувати шифр з блоком заміни розмірністю 16x16 (потрібно 128 Кбайт пам’яті) і довжиною слова 16 біт. Етап ініціалізації буде відбуватися значно повільніше, але в результаті алгоритм все одно буде швидшим.

В алгоритмі RC–6 передбачено використання чотирьох робочих регістрів, а також введена операція цілочисельного множення, яка дозволяє збільшити збурення, створене кожним циклом шифрування, що приводить до збільшення стійкості та можливості зменшити число циклів.

RC–6 є повністю параметризованим алгоритмом шифрування. Конкретна версія RC–6 позначається як RC–6–w/r/b, w позначає довжину слова в бітах, r – ненульова кількість ітераційних циклів шифрування, b – довжина ключа в байтах. У всіх варіантах RC–6–w/r/b працює з чотирма w – бітовими словами, використовуючи шість базових операцій, які позначаються таким чином:

$a+b$ – цілочисельне додавання по модулю 2^w;

$a-b$ – цілочисельне віднімання по модулю 2^w;

$a \oplus b$ – побітове виключаюче або w–бітових слів;

$a \times b$ – цілочисельне множення по модулю 2^w;

a<<b – циклічний зсув w–бітового слова ліворуч на величину, задану $\log_2 w$ молодшими бітами b;

a>>b – циклічний зсув w–бітового слова праворуч на величину, задану $\log_2 w$ молодшими бітами b.

Алгоритм обчислення ключів виглядає таким чином. Користувач задає ключ довжиною b байтів. Достатня кількість ненульових байтів дописується в кінець, щоб отримати ціле число слів. Потім ці байти записуються, починаючи з молодшого, в масив з c слів.

Структура шифру RC–6 є узагальненням мережі Фейстела. Блок тексту розбивається не на 2, а на 4 підблоки і на кожній ітерації змінюються два підблоки з чотирьох. При цьому в кінці ітерації шифрування відбувається циклічний зсув підблоків ліворуч (при розшифруванні відповідно праворуч). Це узагальнення привело до того, що була втрачена властивість інваріантності блоків шифрування і розшифровки, хоча це не є визначальним в оцінці даного алгоритму.

Тема 7. Арифметика асиметричних криптосистем, генерація ключів.

Для будь-якого цілого a і натурального b однозначно визначені цілі числа *q* і r такі, що a = *bq* + *r* і *0* < *r* < *b*. Число q називається *часткою,* а r – *остачею* від ділення a на b Наприклад, рівність –20= (–1)·67 + 47 означає, що –20 при діленні на 67 дає частку –1 і остачу 47. Для остачі будемо вживати таке позначення r = a mod *b*. Число r будемо також називати *(зведеним) лишком* числа a за модулем b. Числа *a* і b називаються *взаємно простими,* якщо НСД (a, b) = 1. *Алгоритм Евкліда* (3 ст. до н.е.) для знаходження НСД двох натуральних чисел a і *b* грунтується на співвідношеннях

НСД (a, b*)* = НСД (a, a mod b*)* для a > b

НСД(a,0) = a

Продемонструємо ідею цього алгоритму на прикладі.

Приклад. Щоб знайти НСД (211, 79), застосовуємо алгоритм Евкліда. Робота алгоритму зводиться до кількаразового ділення з остачею:

$$211 = 79 \cdot 2 + 53$$
$$79 = 53 \cdot 1 + 26$$
$$53 = 26 \cdot 2 + 1$$
$$26 = 1 \cdot 26 + 0$$

Маємо НСД (211, 79) = НСД (79, 53) = НСД (53, 26) = НСД (26,1) = НСД(1,0) = 1.

Алгоритм Евкліда дає такий наслідок.

Твердження 2.2. *Для кожної пари взаємно простих чисел a і b можна знайти такі цілі u і v, що ua + vb* = 1.

Приклад. Нехай a = 211, *b* = 79. Протокол роботи алгоритму Евкліда виписаний у попередньому прикладі. Рухаючись знизу вверх, отримуємо

$1 = 1 \cdot 53 + (-2) \cdot 26 = 1 \cdot 53 + (-2) \cdot (79 - 1 \cdot 53) = (-2) \cdot 79 + 3 \cdot 53 = (-2) \cdot 79 + 3 \cdot (211 - 2 \cdot 79) = 3 \cdot 211 + (-8) \cdot 79.$

Отже, *u* = 3 і *v*= – 8.

Елемент, обернений до x за модулем n відносно множення, будемо позначати через x^{-1} mod n або просто x^{-1}. Це означає, що $x \cdot x^{-1} \bmod n = 1$.

ПРИКЛАД. Нехай ми хочемо знайти елемент, обернений до 79 за модулем 211. Вище була отримана рівність 1 = 3·211 + (– 8·79. З неї випливає, що (–8)·79 = 1(mod211). Отже, $79^{-1} \bmod 211 = (-5) \bmod 211 = 203$.

Функцією Ейлера від числа n є кількість натуральних чисел, взаємно простих з n і позначається ϕ(n).

ТЕОРЕМА ЕЙЛЕРА (1763). *Для взаємно простих цілого x і натурального n справедлива конгруенція* $x^{\phi(n)} = 1 \pmod n$.

МАЛА ТЕОРЕМА ФЕРМА (1640). *Якщо ціле x не ділиться на просте p, то* $x^{p-1} = 1 \pmod p$.

КИТАЙСЬКА ТЕОРЕМА ПРО ОСТАЧІ (І СТ. ДО Н.Е.). *Для будь-якої пари взаємно простих натуральних чисел* n_1 *і* n_2 *та для будь-якої пари цілих чисел* x_1 і x_2, *можна знайти таке ціле x, що* $x = x_1 \pmod{n_1}$ і $x = x_2 \pmod{n_2}$.

ПРИКЛАД. Нехай ми хочемо знайти ціле *x,* яке при діленні на 211 давало б остачу 100, а при діленні на 79 остачу 10. Вище була отримана рівність 1 = 3·211 + (-8)·79. Отже, в якості *x* можна взяти: 10·3·211+ 100·(-8)·79 = -56870. Зрозуміло, що це число можна замінити його остачею від ділення на 211·79 = 16669. В результаті отримуємо 9806.

Тема 8. Криптосистема RSA.

Запропонована 1977 року система RSA є чи не найполулярнішою криптосистемою з відкритим ключем. Назва системи утворена з перших літер імен її винахідників — Рональда Райвеста, Алі Шаміра та Леонарда Адлемана.

Генерування ключів. Вибирають два досить великі прості числа *p* і q. Для їх добутку *n= pq* значення функції Ейлера дорівнює φ(n) = (p–1)(q–1) = *n–p–q*+1. Далі випадковим чином вибирають елемент e, що не перевищує значення *ϕ(n)* і взаємно простий з ним. Для e за алгоритмом Евкліда знаходять елемент *d,* обернений до e за модулем *ϕ(n),* тобто $ed = 1 \pmod{\phi(n)}$.

Як результат покладають:

Відкритий ключ: e, n.

Таємний ключ: d.

Шифрування відбувається блоками. Для цього повідомлення записують у цифровій формі і розбивають на блоки так, що кожен блок позначав число, яке не перевищує *n.* Алгоритм шифрування *E* у системі RSA полягає у піднесенні *M* до степеня e. Записуємо це так: $E(M) = M^e \bmod n$. В результаті отримується блок криптотексту $C = E(M)$.

Дешифрування. Алгоритм дешифрування *D* блоку криптотексту *C* полягає у піднесенні *C* до степеня *d,* тобто

$$D(C) = C^d \bmod n.$$

ПРИКЛАД 2.1. Нехай p = 53 і q = 67. Тоді n = 3551 і φ(n) = 3432. Візьмемо e = 1021 — за допомогою розширеного алгоритму Евкліда легко перевірити, що НСД (1021,3432) = 1. Одночасно обчислюємо d=1021^{-1} mod 3432 = 1237. Ключі вибрано.

Відкритий ключ e = 1021 і n = 3551 оприлюднюємо. Тепер будь-хто може послати нам зашифроване повідомлення. Припустимо, один із ділових партнерів вирішив послати нам вказівку ПРОДАЙ. Спочатку він перетворює своє повідомлення у цифрову форму, замінюючи кожну літеру її двоцифровим десятковим номером в алфавіті: 1920 1805 0013. Видно, що з нашим модулем n цифрове повідомлення варто розбивати на блоки по 4 цифри, як це і зроблено. При шифруванні перший блок 1920 перетворюється у 1920^{1021} mod 3551 = 2393. Таким же чином шифруються наступні два блоки, і в результаті виходить криптотекст 2393 17S8 2188.

Отримавши цей криптотекст, проводимо дешифрування піднесенням кожного блоку до степеня d = 1237 за модулем n = 3551. Можна переконатись, що 2393^{1237} mod 3551 = 1920 і т.д.

Тема 9. Криптосистеми Рабіна та Ель–Гамаля.

Криптосистема Рабіна.

Генерування ключів. Вибирають два великі прості числа p і q. Обчислюють їх добуток $n = pq$. Покладають

Відкритий ключ: n .

Таємний ключ: p, q.

Шифрування відбувається блоками подібно до системи RSA, згідно з формулою

$$E\{M) = M^2 \text{ mod } л.$$

Алгоритм дешифрування складніший, тому розглянемо його на прикладі.

Нехай таємний ключ вибрано так: p= 53 і q = 67. Тоді відкритим ключем буде n = 3551.

Розглянемо шифрування повідомлення ПРОДАЙ. Спочатку повідомлення записується у цифровій формі і розбивається на блоки по чотири цифри: 1920 1805 0013. Поший блок 1920 перетворюється у 1920^2 mod 3551 = 0462. Подібно шифруються наступні два блоки, і в результаті виходить криптотекст: 0462 1758 0169.

Припустимо тепер, що ми отримали криптотекст 1497. Для шифрування слід з нього добути квадратні корені за модулем 3551. З цією метою добуваємо корені за простими модулями 53 і 67 із лишків 1497 mod 53 = 13 і 1497 mod 67 = 23, відповідно. Знаходимо $\sqrt{13}$ mod 53 = 15, 38 і $\sqrt{23}$ mod 67 = 31, 36. За допомогою алгоритму з Китайської теореми про остачі визначаємо чотири корені з 1497 за модулем 3551: (15,31) = 0969, (15,36) = 1711, (38,31) = 1840, (38,36) = 2582. Як зразу видно, лише другий корінь є числовим еквівалентом тексту в українській абетці, а саме повідомлення НІ.

Криптосистема Ель–Гамаля.

Генерування ключів. Вибирають велике просте p, а також просте число g, $1 < g < p - 1$. Ці числа не є таємницею і перебувають в загальному користуванні. Кожен абонент вибирає собі випадкове число a у проміжку від 1 до p–1, і обчислює $h = g^a \bmod p$.

Відкритий ключ: *p, g, h.*
Таємний ключ: *a.*

Шифрування відбувається блоками. Кожен блок *M* не повинен перевищувати p.

- Вибирають випадкове число r таке, що $1 < r < p - 1$.
- Обчислюють $C = (c_1, c_2)$, де

$$c_1 = g^r \bmod p, \quad c_2 = Mh^r \bmod p.$$

Дешифрування. Маючи таємний ключ a і криптотекст $C = (c_1, c_2)$, обчислюють:

$$M = c_2 \cdot (c_1^{-1})^a \bmod p$$

Приклад. Нехай p=23, g=5, a=6. Обчислюємо $h=5^6 \bmod 23=8$. Відкритий і таємний ключ сформовано.

Припустимо, що шифрується числова інформація, і потрібно зашифрувати повідомлення $M = 7$. Нехай вибрано r = 10. Тоді $c_1 = 5^{10} \bmod 23 = 9$ і $c_2 = (7 \cdot 8^{10}) \bmod 23 = 21$. Отримуємо криптотекст $C = (9,21)$. Що стосується дешифрування, то легко перевірити, що справді $D(9,21) = 21 \cdot (9^6)^{-1} \bmod 23 = 7$.

Тема 10. Алгоритми електронного цифрового підпису.

Підпис у системі RSA.

В системі RSA кожен абонент має пару ключів – загальновідомий відкритий і таємний, який знає лише абонент і ніхто інший. Таким чином, будь-хто може скористатися алгоритмом шифрування E_x абонента *X,* але тільки він сам володіє алгоритмом дешифрування D_x. Важливим є виконання таких співвідношень для довільного повідомлення M: Dx(Ex(M)) = Ex(Dx(M)) = M.

Ці співвідношення зводяться до рівностей $(M^{ex})^{dx} = (M^{dx})^{cx} = M$ і виражають той факт, що шифруюче відображення *Ex* та дешифруюче *Dx* є взаємно оберненими.

Припустимо тепер, що абонент А хоче послати абонентові Б повідомлення *M* таким чином, щоб той був певен, що повідомлення справді послане абонентом А. Для цього пропонується такий протокол, в якому (E_A, D_A) та $(E_в, A_в)$ — алгоритми шифрування та дешифрування абонентів А та Б.

- Абонент А обчислює $C = E_B(D_A(M))$ і посилає *C* абонентові Б.
- Абонент Б, отримавши С, обчислює $M = E_A(D_B(C))$.

Коректність протоколу зводиться до рівності

$$E_A(D_B(E_B(D_A(M)))) = M.$$

Підпис у системі Ель–Гамаля.

Для генерування ключів *в*ибирають велике просте p, а також просте число g, $1 < g < p-1$. Числа *p* і g не є таємницею і перебувають в загальному

користуванні. Кожен абонент вибирає собі випадкове число *a* у проміжку від 1 до *p*–1, і обчислює $h = g^a \bmod p$.

Відкритий клюй: p, g, h. Таємний клюй: a.

Підписування. Абонент А виробляє свій підпис *S* на повідомленні *M* таким чином:

- вибирає випадкове число 1<r<p-1;
- обчислює $s_1 = g^r \bmod p$;
- обчислює $r' = r^{-1} \bmod (p - 1)$;
- обчислює $s_2 = (M — as_1)r' \bmod (p — 1)$;
- покладає $S = (s_1, s_2)$.

Підтвердження підпису.

- Абонент Б перевіряє, чи $g^M = h^{S1} s_1^{S2} \pmod p$.

DSA.

Запропонований у 1991 році.

Генерування ключів. Вибирають велике просте число p таке, що p – 1 має досить великий простий дільник *q*. Стандарт вимагає, щоб $2^{512} < p < 2^{1024}$ і $q > 2^{160}$. Далі вибирають довільний елемент *h* порядку *q*. Параметри *p*, *q*, *h* не становлять таємниці і є спільними для всіх абонентів мережі.

Абонент А вибирає випадкове число *a* в діапазоні від 0 до *q* – 1 і обчислює число $b = h^a \bmod p$. Його ключі формуються так.

Відкритий КЛЮЧ: b таке, що $b = h^a \bmod p$.

Таємний ключ: a.

Підписування. Алгоритм підписування використовує вкорочуючу функцію f з довжиною вкорочення 160 бітів. Щоб виробити свій підпис *S* для повідомлення M, абонент А:

- вибирає випадкове число r в діапазоні від 0 до q-1;
- обчислює $r' = r^{-1} \bmod q$;
- обчислює $s_1 = (h^r \bmod p) \bmod q$;
- обчислює $s_2 = (r'(f(M) + as_1)) \bmod q$;
- формує підпис $S = (s_1, s_2)$.

Підтвердження підпису. Отримавши повідомлення *M* із підписом $S = (s_1, s_2)$, абонент Б:

- обчислює $s' = S_2^{-1} \bmod q$;
- обчислює $u_1 = (f(M)s') \bmod q$;
- обчислює $u_2 = (s_1 \cdot s') \bmod q$;
- обчислює $t = (h^{U1} \cdot b^{U2} \bmod p) \bmod q$;
- перевіряє рівність $t = s_1$.

Тема 11. Криптографічні протоколи (обмін ключем, жереб по телефону, розподіл таємниці тощо).

Обмін ключем.

Нехай абоненти А і Б, які, спілкуючись через канал, що ймовірно прослуховується, хочуть домовитися про спільний таємний ключ. Тоді:

- абонент А вибирає велике просте число p та просте 1<g<p-1 і відкрито, не роблячи з цього жодної таємниці, посилає p і g абонентові Б;
- абонент А вибирає випадкове число *a* в межах від 1 до p – 1, а абонент Б – випадкове число *b* в тих же межах;
- абонент А обчислює g^a mod p і посилає це значення абонентові Б, який обчислює g^b mod p і теж посилає абоненту А;
- обидва абоненти обчислюють одне і теж число

$$(g^b)^a \bmod p = (g^a)^b \bmod p = g^{ab} \bmod p,$$

яке і приймають в якості ключа.

ПРИКЛАД. Нехай p = 97, а g = 5. Припустимо, що абонент А вибрав число a = 12, а абонент Б вибрав b = 63. Тоді абонент А посилає абоненту Б 5^{12} mod 97 = 42, абонент Б абоненту А 5^{63} mod 97 = 75, і обоє обчислюють 75^{12} mod 97 = 42^{63} mod 97 = 21.

Жереб по телефону

- абонент А вибирає своє число *a* і пару ключів *(K,K')*. Після цього зашифровує *a* і результат *c = Eк(a)* разом з ключем К посилає абоненту Б;
- абонент Б вибирає число b і посилає його абоненту А;
- абонент А посилає абоненту Б дешифруючий ключ К’;
- абонент Б перевіряє, що *(K,K')* справді є парою шифруючого та дешифруючого ключів, і обчислює a = $D_{K'}(C)$.
- Обоє обчислюють R = a ⊕b.

Гра в карти заочно:

- абоненти А і Б досягають згоди про кодування карт словами M_1,..., M_{52}, і домовляються, яка саме комутативна криптосистема буде використовуватись;
- Обоє таємно один від другого вибирають собі шифруючий та дешифруючий ключі;
- абонент А зашифровує повідомлення M_1,..., M_{52}, перемішує випадковим чином крнптотексти $E_A(M_1)$, ..., $E_A\{M_{52}\}$ і посилає їх абоненту Б;
- абонент Б вибирає випадкові п'ять криптотекстів, і посилає їх назад абоненту А. Це карти, якими буде грати абонент А;
- із карт, що залишилися, абонент Б вибирає ще п'ять для себе;
- абонент Б зашифровує відібрані карти за допомогою власного ключа і отримані криптотексти посилає абоненту А;
- абонент А дешифрує отримані криптотексти і повертає абоненту Б результат;
- абонент Б дешифрує надіслані абонентом А криптотексти і отримує свою п'ятірку карт;
- в кінці гри абоненти обмінюються ключами і перевіряють, чи ніхто з них не хитрував.

Розподіл таємниці.

Нехай натуральне число *s* є цінною секретною інформацією (номер рахунку у швейцарському банку, код команди на запуск балістичної ракети

тощо). Завданням протоколу є так подрібнити секрет s на частини, по одній для кожного із *n* учасників, щоб будь-які *k* учасників могли відновити s, поєднавши свої частинки, але щоб ніяка група з *k* – 1 учасника цього зробити не могла.

Вибирають досить велике просте число p, більше за s. Покладають a_0 = s, і вибирають випадковим чином числа a_1,...., a_{k-1}. Нехай $f(x) = \Sigma_{0\le i<k} a_i x^i$ – многочлен від змінної *x*. і-ий учасник протоколу, де 1 ≤i≤ n, отримує значення s_i=f(i).

Якщо відомі довільні *к* значень $f(i_1)$, . . ., $f(i_k)$, то многочлен f можна реконструювати за інтерполяційною формулою Лагранжа:

$$f(x) = \sum_{i=1}^{k} f(i_1) \prod_{j \neq 1} \frac{x - i_j}{i_i - i_j}.$$

Після цього легко знаходиться секрет $s = a_0$ = f(0).

Знання лише *к* - 1 значення функцій f не дає *жодної* інформації про секрет.

Тема 12. Класичні та сучасні методи криптоаналізу.

Шифр заміни над n-символьним алфавітом має *n!* ключів. Для значень *n*= 26, 33 (латинський та український алфавіти) це число є дуже великим. Для його оцінки можна скористатися варіантом формули Стірлінга, звідки для *n* = 26 отримуємо $n > 10^{26}$. Число справді велике — нагадаємо, що наша планета існує лише 10^9 років, а наступний льодовиковий період очікується через 14000 років, тобто $4{,}41504 \cdot 10^{11}$ секунд. Це співставлення переконливо засвідчує безперспективність брутальної атаки на шифр заміни, однак цього недостатньо аби стверджувати, шо він є надійним. Виявляється, успішний криптоаналіз можливий за допомогою *частотного методу.*

Частота символу у тексті дорівнює кількості його входжень у цей текст, поділеній на загальну кількість букв у тексті. Наприклад, частота букви а у тексті «купила мама коника» дорівнює 4/18 = 2/9, а частота пропуску між словами у цьому ж тексті дорівнює 2/18 = 1/9. Для кожної мови справджується такий емпіричний факт:

> *У досить довгих текстах кожна буква зустрічається із приблизно однаковою частотою, залежною від самої букви і незалежною від конкретного тексту.*

На підставі цього факту із кожним символом пов'язують деяке число, *частоту цього символу в мові,* з якою приблизно він зустрічається в кожному великому тексті цією мовою. Підрахунок частот символів у мові здійснюють на основі вибраного навмання середньостатистичного тексту. Відповідно до цього складені таблички частот для різних мов.

Припустимо, що перехоплено довгий криптотекст (або послідовність багатьох коротких), отриманий за допомогою шифру заміни. Частотним методом можна здійснити дешифрування, навіть не знаючи ключа. Для niv: го обчислюють частоти кожного символу в криптотексті і порівнюють отримані результати з табличкою частот для мови, якою написане повідомлення. Не слід сподіватися, що таким чином можна буде однозначно встановити ключ, але

перебір це дозволить скоротити радикально. Наприклад, якщо при шифруванні не ігноруються пропуски між словами, то найпоширеніший символ у криптотексті поза сумнівом відповідає саме пропуску. А відтак стає відомою сукупність символів, що відповідають словам з одної букви (в українській мові це а, б, в, б, ж, з, і, й, о, у, я) та словам з двох букв (це, не, на, до та інші), що дозволяє ці символи розпізнати ціною справді невеликого перебору. Свою роль при частотному аналізі відіграє та обставина, що кожна мова володіє властивістю *надлишковості,* тобто текст можна поновити навіть коли частина його букв невідома.

Миттєвою є користь від частотного аналізу при розкритті шифру зсуву. Проілюструємо загальну ідею таким прикладом. Нехай нам належить розшифрувати криптотекст пцпспофнпмплпибгпефрптмбвмєоп, який був отриманий шифром зсуву, причому пропуски та розділові знаки ігнорувались. Підраховуємо частоти і зауважуємо, що найбільша, а саме 9/29, припадає на літеру п. Природньо припустити, що у відкритому тексті їй відповідає найпоширеніша в українській мові літера о. Це означало б, що довжина зсуву дорівнює 1. Виконуємо обернений зсув, тобто на одну позицію вліво, і справді отримуємо змістовне повідомлення, що охоронумолокозаводупослаблено.