Лекція №8 (НЕ 2.1, 2 год.). Симетричні криптосистеми.

План лекції:

1. Особливості стандарту ГОСТ 28147-89.
2. Особливості стандарту AES.

Зміст лекції:

1. Особливості стандарту ГОСТ 28147-89.

Стандарт шифрування даних Радянського Союзу, ГОСТ 28147-89, було прийнято у 1989 році. В основі цього стандарту лежить DES-подібний алгоритм симетричного шифрування.

Розробники вказаного стандарту могли врахувати всі недоліки DES і успішно це зробили. ГОСТ 28147-89 й сьогодні залишається одним із найстійкіших алгоритмів шифрування.

Авторам цієї книжки не відомі успішні атаки на цей алгоритм. Його і досі успішно використовують на теренах колишнього Радянського Союзу для шифрування даних різного ступеня секретності.

ГОСТ є 64-бітовим алгоритмом із 256-бітовим ключем. У процесі роботи алгоритму на 32 етапах послідовно виконується простий алгоритм шифрування. Для шифрування текст спочатку розбивається на ліву половину L і праву половину R. На етапі і використовується підключ K_i. На і-му етапі алгоритму ГОСТу виконується таке:

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

Етап ГОСТу показаний на рисунку. Функція f досить проста. Спочатку права половина додається за модулем 2^{32} з і-им підключем. Результат розбивається на вісім 4-бітових частин, кожна з яких надходить на вхід свого S-блока. ГОСТ використовує вісім різних S-блоків, перші 4 біти потрапляють у перший S-блок, другі 4 біти - в другий S-блок і т. д. Кожен S-блок є перестановкою чисел від 0 до 15. Наприклад, S-блок може виглядати так:

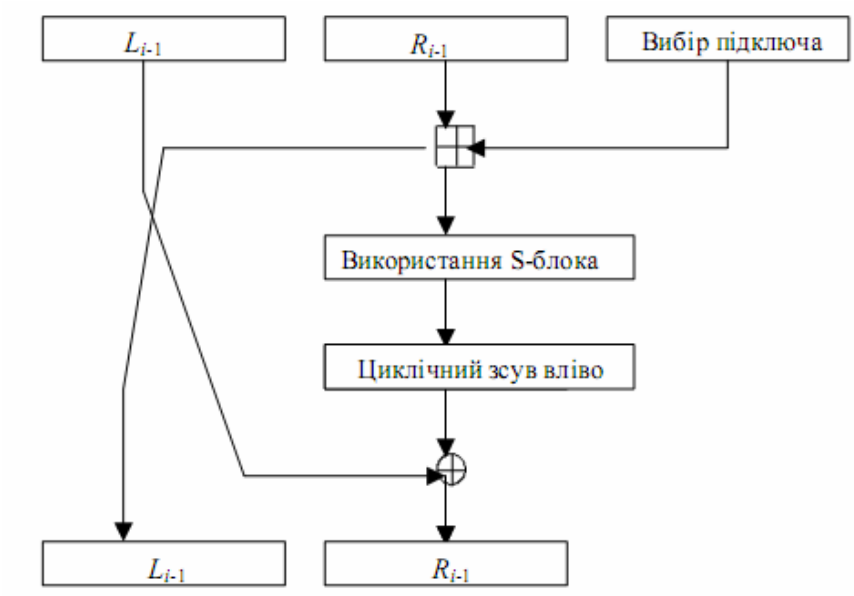
7, 10, 2, 4, 15, 9, 0, 3, 6, 12, 5, 13, 1, 8, 11.

У цьому випадку, якщо на вході S-блока 0, то на виході 7. Якщо на вході 1, на виході 10 і т. д. Усі вісім S-блоків різні, вони фактично є додатковим ключовим матеріалом. S-блоки зберігаються в таємниці.

Виходи всіх восьми S-блоків об'єднуються в 32-бітове слово, потім все слово циклічно зсувається вліво на 11 бітів. Нарешті результат об'єднується з допомогою XOR із лівою половиною, й отримується нова права половина, а права половина стає новою лівою половиною. Виконавши це 32 рази, отримаємо результат.

Генерація підключів досить проста. 256-бітовий ключ розбивається на вісім 32-бітових блоків: k1, k2, …, k8. На кожному етапі використовується свій підключ, як показано в табл. 3.9. Дешифрування виконується так само, як і шифрування, але інвертується порядок підключів ki.

Стандарт ГОСТу визначає не спосіб генерації S-блоків, а лише спосіб їх зображення. Виробник створює перестановки S-блока самостійно за допомогою генератора випадкових чисел. Час від часу S-блоки повинні змінюватися.



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

Порівняння крипостійкості DES і ГОСТ

Ось головні відмінності між DES і ГОСТом:

- DES використовує складну процедуру для генерації підключів із ключів. У ГОСТ ця процедура дуже проста;
- у DES 56-бітовий ключ, а в ГОСТі ñ 256-бітовий. Якщо додати секретні перестановки S-блоків, то повний обíєм секретної інформації ГОСТу складе приблизно 610 бітів;
- у S-блоків DES 6-бітові входи і 4-бітові виходи, а у S-блоків ГОСТу 4-бітові і входи і виходи. В обох алгоритмах використовується по вісім S-блоков, але розмір S-блока ГОСТу дорівнює одній четвертій розміру S-блока DES;
- у DES використовуються нерегулярні перестановки, названі P-блоком, а в ГОСТі - 11-бітовий циклічний зсув вліво;
- у DES 16 етапів, а в ГОСТі - 32.

Якщо найкращим способом зламу ГОСТу є метод «грубої сили», то це дуже безпечний алгоритм. ГОСТ використовує 256-бітовий ключ, а якщо враховувати секретні S-блоки, то довжина ключа зростає. ГОСТ, очевидно, стійкіший до диференційного і лінійного криптоаналізу, ніж DES. Хоча випадкові S-блоки ГОСТу слабкіші від фіксованих S-блоків DES, їх секретність збільшує стійкість ГОСТ до диференційного і лінійного криптоаналізу. До того ж ці способи зламу вразливі до кількості етапів ñ чим більше етапів, тим важчий злам. ГОСТ використовує в два рази більше етапів, ніж DES. Саме це робить неспроможними і диференційний, і лінійний криптоаналізи.

Інші частини ГОСТ такі ж, як у DES, або навіть слабкіші. ГОСТ не використовує перестановку з розширенням, на відміну від DES. Видалення цієї перестановки з DES послаблює його через зменшення лавинного ефекту, тому вважається, що відсутність такої операції в ГОСТі послаблює цей алгоритм. Додавання, що використовується в ГОСТі, не менш безпечне, ніж використовувана в DES операція XOR.

Найбільшою відмінністю є використання в ГОСТі циклічного зсуву замість перестановки. Перестановка DES збільшує лавинний ефект. У ГОСТі зміна одного вхідного біта впливає на один S-блок одного етапу, який потім впливає на два S-блоки наступного етапу, три блоки наступного етапу і т.д. У ГОСТі потрібно 8 етапів, перш ніж зміна одного вхідного біта вплине на кожен біт результату, алгоритму DES для цього потрібно тільки 5 етапів. Це, звичайно ж, слабке місце. Але ГОСТ складається з 32 етапів, а DES ñ тільки з 16. Розробники ГОСТу намагалися досягти рівноваги між безпекою і ефективністю. Вони змінили ідеологію DES так, щоб створити алгоритм, який більше підходить для програмної реалізації. Вони, можливо, менш упевнені в безпеці свого алгоритму і спробували компенсувати це дуже великою довжиною ключа, збереженням у секреті S-блоков і подвоєнням кількості ітерацій.

2.Особливості стандарту AES

У кінці 1996 р. Національним інститутом стандартів США (NIST) було оголошено конкурс на створення нового загальнонаціонального стандарту шифрування, який повинен прийти на заміну DES. Розробленому стандарту присвоїли робочу назву AES (Advanced Encryption Standard).

2 жовтня 2000 року прийнято остаточне рішення. Як запропонований стандарт обрали алгоритм Rijndael. Цей алгоритм розроблений Вінсентом Рійманом і Йоаном Дайменом і є алгоритмом, який не використовує сітку Фейстеля.

На відміну від DES і ГОСТу, алгоритм Rijndael використовує кінцеву групу точок еліптичної кривої.

Із теорії чисел відомі операції цілочисельного ділення та залишку від ділення. Наприклад, 7:5 = 1 і 2 у залишку. Взяття залишку від цілочисельного ділення записується так: 2 = 7mod5. Залишки за mod p утворюють кінцеву послідовність цілих чисел від 0 до p-1. Наприклад, група залишків за mod 7 складається з чисел 0, 1, 2, 3, 4, 5, 6.

Опис AES

В алгоритмі AES використовується поле Галуа, побудоване як розширення поля за коренями деякого многочленна. Цей многочлен обрано з міркувань ефективності представлення елементів поля. Елементарні операції, які використовуються в алгоритмі, виконуються у вказаному полі. Алгоритм Rijndael являє собою блоковий шифр зі змінною довжиною блока і змінною довжиною ключа. Довжини блока і ключа можуть бути обрані незалежно такими, що дорівнюють 128, 192 чи 256 бітам. 256 бітів алгоритму Rijndael базується на прямому перетворенні блока, який зображається як матриця байтів. на кожному кроці обробляється блок в цілому й незмінних частин блока не залишається. Оскільки за один раунд шифрується новий блок, таких раундів треба менше. Блок-схему алгоритму подано на рисунку. Вхідні дані M подаються у вигляді прямокутної матриці байтів $n \times 4$, де n=4,6,8 в залежності від розміру блока. Позаяк AES використовує розмір блоку в 128 бітів, то матриця байтів має розмір 4×4.

Раундова функція F складається з чотирьох етапів:

1) BS (Byte Sub) - побайтова заміна;
2) SR (Shift Row) - зсув рядків;
3) MC (Mix Column) - операція над стовпчиками, коли кожен стовпчик множиться за правилом поля Галуа на точках еліптичної кривої на фіксовану матрицю.
4) AK (Add Round Key) - додавання раундового ключа.

Додавання ключа ітерації виконується шляхом простого побітового додавання за модулем 2 кожного байта масиву з відповідним байтом ключа. Це перетворення обернене самому собі.

Алгоритм обробки ключа

Ключі ітерації отримуються з ключа шифрування з допомогою алгоритму обробки ключа, який складається з двох компонентів - розширення ключа й вибору ключа ітерації. Основними принципами його побудови є:

1. Загальна кількість біт ключів ітерації дорівнює довжині блока, помноженій на кількість ітерацій плюс одиниця (наприклад, для блока розміром 128 бітів і 10 ітерацій потрібно 1408 бітів ключів ітерації).
2. Ключ шифрування збільшується до розширеного ключа.
3. Ключі ітерації беруться з розширеного ключа в такий спосіб: перший ключ ітерації складається з перших Nb слів, другий - із наступних Nb слів і т.д.

Криптостійкість алгоритму Rijndael

Дослідження показали, що криптоаналіз 16-раундового DES у принципі реальний, однак вимагає великої кількості вихідних даних. Криптоаналіз при 20-24 циклах стає навіть теоретично неможливим.

Алгоритм ГОСТу 28147-89 має 32 раунди. Як ми бачили, його криптостійкість має значний запас і сьогодні. У відкритих публікаціях відсутні повідомлення про успішні атаки на ГОСТ.

За оцінками розробників Rijndael, вже на чотирьох раундах шифрування цей алгоритм досить стійкий для сучасних використань. Теоретичною межею вважається 6-8 раундів. Отже, 10-14 раундів, передбачених у цьому алгоритмі, мають значний запас щодо криптостійкості.

До недоліків алгоритму можна віднести нетрадиційну, недостатньо вивчену схему, яка може містити приховані вразливості, невідомі на час конструювання алгоритму.

Насамкінець, подамо порівняльну таблицю характеристик блокових алгоритмів.

Характеристика	**DES**	**ГОСТ**	**IDEA**	**Rijndael**
Довжина блока за 1 раунд, біти	64	64	64	128, 192, 256
Довжина ключа, біти	64 (56)	256	128	128, 192, 256
Розмір раундового ключа, біти	48	32	16	128, 192, 256
Операції	Адитивні підстановки та зсуви		Додавання та множення	Операції над кінцевими полями
Еквівалентність прямого і обернених перетворень	До порядку розміщення ключових елементів		До використання ключового елемента	До вектора ключових елементів, вузла замін та інших констант
Кількість раундів	16	32	8	10, 12, 14

Як стандарт AES було обрано варіант Rijndael із розміром блока у 128 бітів. Кількість раундів залежить, як бачимо з таблиці, від розміру блока й ключа. Якщо максимальний із них дорівнює 128 бітам, то використовують 10 раундів, якщо 192 бітам - 12, для 256 бітів - 14 раундів. Це дозволяє обирати оптимальний варіант у залежності від потрібної криптостійкості та швидкості обробки.