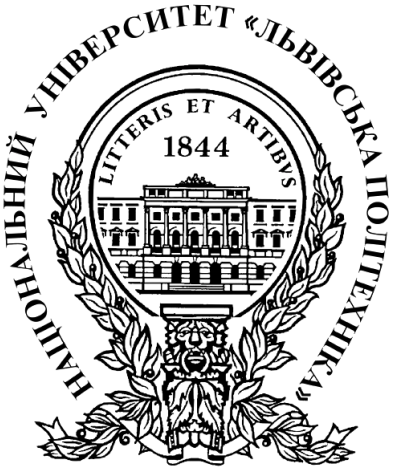
Міністерство Освіти і Науки України

Національний Університет "Львівська Політехніка"



Кафедра ЕОМ

Методичні вказівки до циклу лабораторних робіт

з курсу "Захист інформації в комп'ютерних системах "
для студентів базового напрямку 6.0915 "Комп'ютерна інженерія"

Затверджено
на засіданні кафедри
Електронних Обчислювальних Машин
Протокол № ____ від _____ 2009 року

Львів – 2009

Лабораторна робота № 1

Шифр моноалфавітної заміни (шифр Цезаря)

Мета роботи: ознайомитись з основами класичної техніки шифрування – шифрами моноалфавітної заміни та типовим прикладом шифрів даного виду - шифром Цезаря.

Теоретичні відомості

При використанні моноалфавітної заміни окремі букви відкритого тексту заміняються іншими буквами або числами або якимись іншими символами. Якщо відкритий текст розглядається як послідовність бітів, то підстановка зводиться до заміни заданих послідовностей бітів відкритого тексту заданими послідовностями бітів шифрованого тексту.

Найдавнішим і найпростішим з відомих підстановочних шифрів є шифр, що використовувався Юлієм Цезарем. У шифрі Цезаря кожна буква алфавіту замінюється буквою, що перебуває на три позиції далі в цьому ж алфавіті. Найпростіше побачити це на прикладі.

Відкритий текст: meet me after the toga party Шифрований текст: PHHW PH DIWHU WKH WRJD SDUMB

Зверніть увагу на те, що алфавіт вважається "циклічним", тому після Z іде A. Визначити перетворення можна, перелічивши всі варіанти, як показано нижче.

Відкритий текст:

Шифрований текст:

a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z
D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C

Якщо кожній букві призначити числовий еквівалент (a = 1, b = 2 і т.д.), то алгоритм можна виразити наступними формулами. Кожна буква відкритого тексту p заміняється буквою шифрованого тексту З:

$$C = E(p) = (p + 3) \bmod(26) .$$

У загальному випадку зміщення може бути будь-яким, тому узагальнений алгоритм Цезаря записується формулою

$$C = E(p) = (p + k) \bmod(26) ,$$

де **к** приймає значення в діапазоні від 1 до 25. Алгоритм дешифрування також простий:

$$p = D(C) = (C - k) \bmod(26) .$$

Якщо відомо, що певний текст був шифрований за допомогою шифру Цезаря, то за допомогою простого перебору всіх варіантів розкрити шифр дуже просто - для цього досить перевірити 25 можливих варіантів ключів. На рис. 1 показані результати застосування цієї стратегії до зазначеного вище повідомлення. У цьому випадку відкритий текст розпізнається в третьому рядку.

Застосування методу послідовного перебору всіх можливих варіантів виправдано наступними трьома важливими характеристиками даного шифру.

1. Відомі алгоритми шифрування й дешифрування.
2. Необхідно перебрати всього 25 варіантів.
3. Мова відкритого тексту відома і легко пізнавана.

Вказівки щодо реалізації:

Шифр Цезаря реалізує кодування повідомлення шляхом «зсуву» усіх символів повідомлення на певне число кк (в оригінальному шифрі дане число дорівнювало 3).

Якщо буква вихідного повідомлення мала позицію jj у вихідному алфавіті, то в зашифрованому повідомленні вона буде замінюватися буквою, що знаходиться на позиції кк+jj. Нехай маємо вихідне повідомлення: “I remember that september”. Будемо використовувати латинський алфавіт зі стандартною послідовністю слідування символів в алфавіті. Результати шифрування наведені в таблиці нижче:

1	i		r	e	m	e	m	b	e	r		t	h	a	t		s	e	p	t	e	m	b	e	r
2	9	0	18	5	13	5	13	2	5	18	0	20	8	1	20	0	19	5	16	20	5	13	2	5	18
3	12	3	21	8	16	8	16	5	8	21	3	23	11	4	23	3	22	8	19	23	8	16	5	8	21
4	12	3	21	8	16	8	16	5	8	21	3	23	11	4	23	3	22	8	19	23	8	16	5	8	21
5	l	c	u	h	p	h	p	e	h	u	c	w	k	d	w	c	v	h	s	w	h	p	e	h	u

Пояснення до таблиці:

1-й рядок - фраза для шифрування;

2-й рядок - номера букв фрази для шифрування в латинському алфавіті;

3-й рядок - номера букв фрази для шифрування, збільшені на 3;

4-й рядок - результат «ділення по модулю 26» чисел 3-го рядка;

5-й рядок - зашифрована фраза

Спроектуємо форму додатку, що дозволить нам вводити повідомлення для шифрування, крок та виводити зашифроване повідомлення:

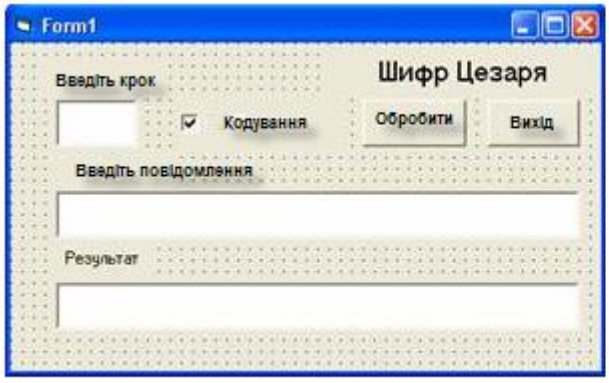


Рис. 1.

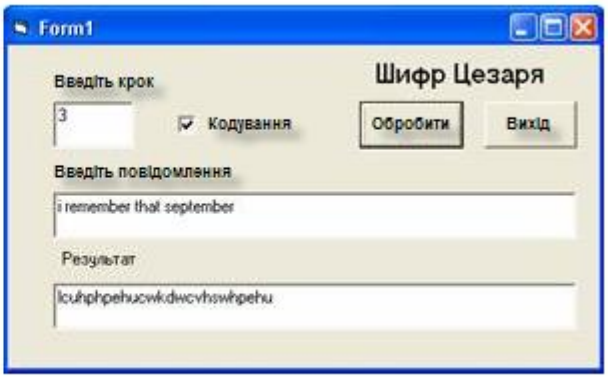


Рис. 2.

Завдання:

1. Створити програму, що реалізує шифрування вихідного повідомлення за допомогою шифру Цезаря з врахуванням того, що повідомлення, що необхідно зашифрувати, написано українською мовою.
В якості кроку шифрування обирається номер студента у списку групи.

2. Оформити і захистити звіт.

Лабораторна робота № 2

Перестановочний шифр

Мета роботи: ознайомитись з основами класичної техніки шифрування – шифрами моноалфавітної заміни та типовим прикладом шифрів даного виду - шифром Цезаря.

Теоретичні відомості

Розглянутий вище метод ґрунтувався на заміщенні символів відкритого тексту різними символами шифрованого тексту. Принципово інший клас перетворень будується на використанні перестановок букв відкритого тексту. Шифри, створені за допомогою перестановок, називають перестановочными шифрами.

Найпростіший з таких шифрів використає перетворення "драбинки", що полягає в тім, що відкритий текст записується уздовж похилих рядків певної довжини ("сходів"), а потім зчитується построчно по горизонталі. Наприклад, щоб шифрувати повідомлення "meet me after the toga party" по методу драбинки зі сходами довжиною 2, запишемо це повідомлення у вигляді

```
m e m a t r h t g p r y
 e t e f e t e o a a t
```

Шифроване повідомлення буде мати такий вигляд.

```
MEMATRHTGPRYETEFETEOAAT
```

Такий "шифр" особливої складності для криптоаналізу не представляє. Більше складна схема припускає запис тексту повідомлення в горизонтальні рядки однакової довжини й наступне зчитування тексту стовпець за стовпцем, але не один по одному, а відповідно до деякої перестановки стовпців. Порядок зчитування стовпців при цьому стає ключем алгоритму. Розглянемо наступний приклад.

Ключ:

```
4 3 1 2 5 6 7
a t t a c k p
o s t p o n e
d u n t i l t
w o a m x y z
```

Відкритий текст:

Шифрований текст: TTNAAPTMTSUOAODWCOIXKNLYPETZ

Простий перестановочный шрифт дуже легко розпізнати, тому що букви в ньому зустрічаються з тією же частотою, що й у відкритому тексті. Наприклад, для тільки що розглянутого способу шифрування з перестановкою стовпців аналіз шифру виконати досить просто - необхідно записати шифрований текст у вигляді матриці й перебрати можливі варіанти перестановок для стовпців. Можна використати також таблиці значень частоти биграмм і триграмм.

Перестановочный шифр можна зробити істотно більше захищеним, виконавши шифрування з використанням перестановок кілька разів. Виявляється, що в цьому випадку застосовану для шифрування перестановку відтворити вже не так просто. Наприклад, якщо попереднє повідомлення шифрувати ще раз за допомогою того ж самого алгоритму, то результат буде наступним.

Ключ:

```
4 3 1 2 5 6 7
t t n a a p t
m t s u o a o
d w c o i x k
n l y p e t z
```

Відкритий текст:

Шифрований текст: NSCYAUOPTTWLTMDNAOIEPAXTTOKZ

Щоб наочніше представити те, що ми одержимо в підсумку повторного застосування перестановки, зіставимо кожну букву вихідного відкритого тексту з номером відповідної їй позиції. Наше повідомлення складається з 28 букв, і вихідною послідовністю буде послідовність

```
01 02 03 04 05 06 07 08 09 10 11 12 13 14
15 16 17 18 19 20 21 22 23 24 25 26 27 28
```

Після першої перестановки одержимо послідовність, що усе ще зберігає деяку регулярність структури.

```
03 10 17 24 04 11 18 25 02 09 16 23 01 08
15 22 05 12 19 26 06 13 20 27 07 14 21 28
```

Після другої перестановки виходить наступна послідовність.

```
17 09 05 27 24 16 12 07 10 02 22 20 03 25
15 13 04 23 19 14 11 01 26 21 18 08 06 28
```

Регулярність цієї послідовності вже зовсім не проглядається, тому її криптоанализ буде вимагати значно більших зусиль.

Завдання:

1. Створити програму, що реалізує довільний перестановочний шифр.
2. Підготувати і захистити звіт, в якому обов’язково навести алгоритм роботи даного перестановочного шифру.

Лабораторна робота №3

Криптоаналіз шифрів моноалфавітної заміни

Мета роботи: ознайомитись з основними методами, що використовуються для криптоаналізу шифрів моноалфавітної заміни та , зокрема, з основами частотного аналізу шифрованого тексту.

Теоретичні відомості

При наявності всього 25 можливих варіантів ключів шифр Цезаря далекий від того, щоб вважатися надійно захищеним. Істотного розширення простору ключів можна домогтися, дозволивши використання довільних підстановок. Давайте ще раз згадаємо шифр Цезаря.

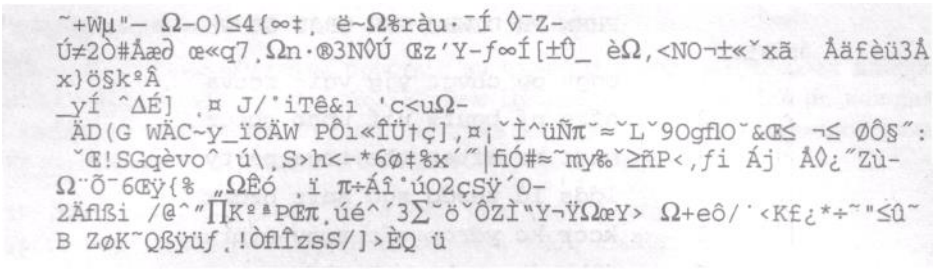


Рис. 1 Приклад стиснутого тексту

Відкритий текст:

a b c d e f g h i j k l m n o p q r s t u v w x y z

Шифрований текст:

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

Якщо в рядку "Шифрований текст" допустити використання кожної з перестановок 26 символів алфавіту, то ми одержимо 26!, або більш ніж 4 x 1026 можливих ключів. Це на 10 порядків більше, ніж розмір простору ключів DES, і це здається достатнім для того, щоб унеможливити успішне застосування криптоаналізу на основі методу послідовного перебору.

Однак для криптоаналітика існує й інша лінія атаки. Якщо криптоаналітик має уявлення про природу відкритого тексту (наприклад, про те, що це нестиснутий текст англійською мовою), можна використати відому інформацію про характерні ознаки, властивим текстам відповідною мовою. Щоб показати, як цей підхід застосовується на практиці, розглянемо невеликий приклад. Припустимо, нам потрібно розшифрувати наступний шифрований текст.

UZQSOVUOHXMOPVGPOZPEVSGZWSZOPFPESXUDBMETSXAIZ
VUEPHZHMDZSHZOWSFPAPPDTSVPQUZWYMXUZUHSX
EPYEPOPDZSZUFPOMBZWPFUPZHMDJUDTMOHMQ

На першому етапі можна визначити відносну частоту появи в тексті різних букв і порівняти їх із середньостатистичними даними для букв англійської мови, показаними на рис. 2..

Якщо повідомлення досить довге, цієї методики вже може бути досить для розпізнавання тексту, але в нашому випадку, коли повідомлення невелике, точного відповідності очікувати не доводиться. У нашому випадку відносна частота входження букв у шифрованому тексті (у відсотках) виявляється наступною.

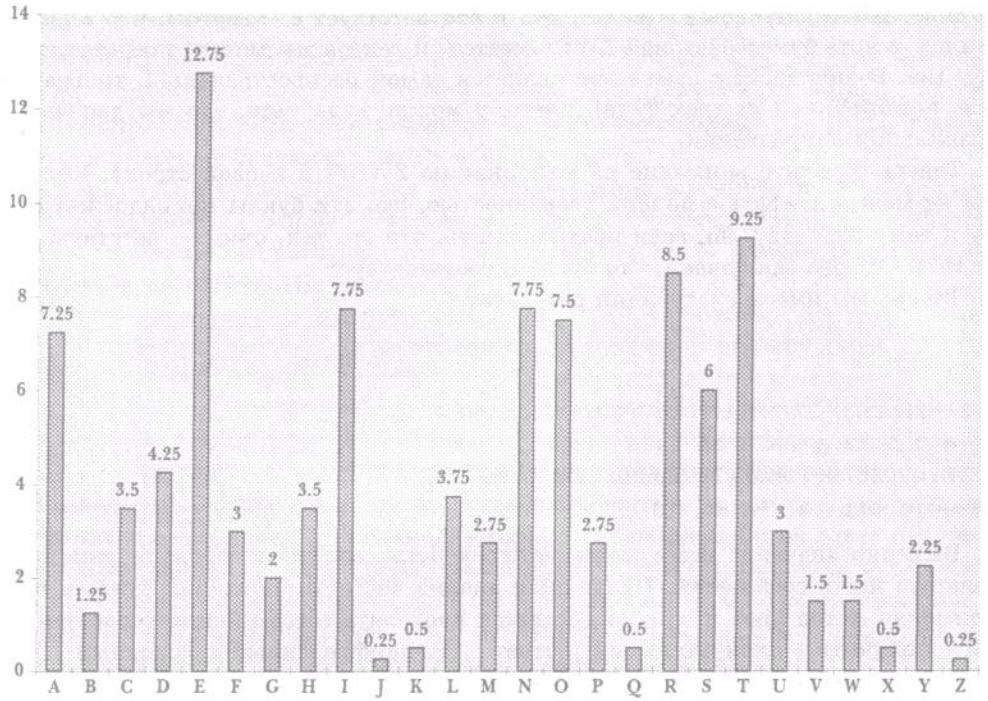


Рис. 2. Відносна частота появи букв в англійському тексті

Порівнюючи ці результати з даними, показаними на рис. 2, можна припустити, що, швидше за все, букви P и Z шифрованого тексту є еквівалентами букв e и t відкритого тексту, хоча важко сказати, якій саме букві - P або Z - відповідає e, а який - t. Букви S, U, ПРО, M и H, що володіють відносно високою частотою появи в тексті, швидше за все, відповідають буквам з множини {r, n, i, o, a, s}.. Букви з низькою частотою появи (а саме A, B, G, Y, I, J), очевидно, відповідають буквам множини {w,v,b,k,x,q,j,z}.

Далі можна піти декількома шляхами. Можна, наприклад, прийняти якісь припущення про відповідності й на їхній основі спробувати відновити відкритий текст, щоб побачити, чи виглядає такий текст схожим на що-небудь змістовне. Більш систематизований підхід полягає в продовженні пошуку в тексті нових характерних закономірностей. Наприклад, може бути відомо, що в розглянутому тексті обов'язково повинні бути присутнім деякі слова. Або ж можна шукати повторювані послідовності букв шифрованого тексту й намагатися визначити їхні еквіваленти у відкритому тексті.

Один з дуже ефективних методів полягає в підрахунку частоти використання комбінацій, що складаються із двох букв. Такі комбінації називають біграмами. Для значень відносної частоти появи в тексті біграмм теж можна побудувати гістограму, подібну показаної на рис. 2. Відомо, що в англійській мові найпоширенішою є біграма th. У нашому шифрованому тексті найчастіше (три рази) зустрічається комбінація ZW. Тому можна припустити, що Z відповідає t, а W - h. Тоді зраніше сформованої гіпотези випливає, що P відповідає e. Помітимо, що в шифрованому тексті буквосполучення ZWP є, і тепер ми можемо представити його як the. В англійській мові the є найпоширенішою триграмою (тобто комбінацією із трьох букв), тому можна сподіватися, що ми рухаємося в правильному напрямку.

Тепер зверніть увагу на комбінацію ZWSW у першому рядку. Звичайно, ми не можемо сказати з повною впевненістю, що ці букви належать тому самому слову, але, якщо припустити, що це так, вони відповідають слову th?t. Звідси робимо висновок, що букві S відповідає a.

Тепер ми маємо наступний результат.

```
UZQSOVUOHXMOPVGPOZPEVSGZWSZOPFPESXUDBMETSXAIZ
 t a        e  e te  a that e e a       a   t
VUEPHZHMDZSHZOWSFPAPPDTSVPQUZWYMXUZUHSX
   e t   ta t ha e ee  a e  th    t  a
EPYEPOPDZSZUFPOMBZWPFUPZHMDJUDTMOHMQ
 e  e e tat  e   the   t
```

З'ясувавши значення всього лише чотирьох букв, ми розшифрували вже значну частину повідомлення. Продовжуючи аналіз частоти появи букв, а також застосовуючи метод «проб і помилок», залишається проробити зовсім небагато роботи, щоб отримати остаточну відповідь. Розшифрований вихідний текст (з доданими в нього пробілами) має такий вигляд.

```
it was disclosed yesterday that several informal but
direct contacts have been made with political
representatives of the viet cong in moscow
```

Моноалфавітні шифри легко розкриваються, тому що вони успадковують частоту вживання букв оригінального алфавіту. Контрзаходом у цьому випадку є застосування для однієї букви не одного, а декількох замінників (називаних омофонами). Наприклад, букві е вихідного тексту може відповідати кілька різних символів шифру, (скажемо, 16, 74, 35 й 21), причому кожен такий символ може використовуватися або по черзі, або за випадковим законом. Якщо число символів-замінників, призначених букві, вибрати пропорційним частоті появи цієї букви, то підрахунок частковості вживання букв у шифрованому тексті стає безглуздим. Великий математик Карл Фрідріх Гаусс (Carl Friedrich Gauss) був упевнений, що з використанням омофонів він винайшов шифр, що неможливо зламати. Але навіть при вживанні омофонів кожному елементу відкритого тексту відповідає тільки один елемент шифрованого тексту, тому в останньому як і раніше повинні спостерігатися характерні показники частоти повторення комбінацій декількох букв (наприклад, біграм), і в результаті завдання криптоаналізу як і раніше залишається досить елементарним.

Щоб у тексті, шифрованому за допомогою методів підстановок, структура вихідного тексту проявлялася менш помітно, можна використати два принципово різних підходи. Один з них полягає в заміщенні не окремих символів відкритого тексту, а комбінацій декількох символів, а інший підхід передбачає використання для шифрування декількох алфавітів.

Завдання:

1. Отримати у викладача згідно з варіантом приклад текстового файлу, зашифрованого шифром Цезаря.
2. Створити програму, що реалізує метод крипто аналізу на основі частотного аналізу шифрованого тексту.
3. Підготувати і захистити звіт.

Лабораторна робота №4

Організація стегоканалу в BMP-файлі

Мета роботи: ознайомитися з поняттям стеганографії та проаналізувати можливості організації стегоканалу в BMP-файлі

Теоретичні відомості

Стеганографія - метод передачі інформації, що приховує саме факт самої передачі інформації. Головна відмінність стеганографії від криптографії, де криптограф точно може визначити чи є передане повідомлення зашифрованим текстом, полягає в можливості вбудовувати секретні повідомлення у відкриті повідомлення так, щоб неможливо було запідозрити існування вбудованого таємного послання. Слово "стеганография" у перекладі із грецького буквально означає "тайнопис" (steganos - секрет, таємниця; graphy - запис). Стеганографія містить у собі безліч секретних засобів зв'язку, таких як невидиме чорнило, мікрофотознімки, умовне розташування знаків, таємні канали й засоби зв'язку наплаваючих частотах і т.д.

Бурхливий розвиток обчислювальної техніки й нових каналів передачі інформації сприяє появі нових стеганографічних методів, в основі яких лежать особливості подання інформації в комп'ютерних файлах, обчислювальних мережах і т.п. Тому останнім часом з'явився новий напрямок стеганографії - комп'ютерна стеганографія.

Комп'ютерна стеганографія - це приховання повідомлення або файлу в іншому повідомленні або файлі. Наприклад, стеганографи можуть сховати аудіо-або відеофайл в іншому інформаційному або навіть у великому графічному файлі. Процес стеганографії можна розділити на кілька етапів.

1. Вибір інформаційного файлу. Першим етапом у процесі стеганографії є вибір файлу, якому необхідно сховати. Його ще називають інформаційним файлом.

2. Вибір файлу-контейнера. Другим етапом у процесі стеганографії є вибір файлу для приховання інформації. Його ще називають файлом-контейнером. У більшості відомих програм по стеганографії говориться, що для приховання інформації, обсяг пам'яті файлу-контейнера повинен десь у вісім разів перевищувати обсяг пам'яті інформаційного файлу. Отже, щоб сховати файл розміром 710КБ, знадобиться графіка обсягом 5600КБ.

3. Кодування файлу. Після того, як обраний інформаційний файл, файл-контейнер і програмне забезпечення по стеганографії, необхідно встановити захист нового файлу по паролю.

4. Відправлення прихованого повідомлення по електронній пошті і його декодування. Останнім етапом у процесі стеганографії є відправлення захованого файлу по електронній пошті і його наступна розшифровка.

Основні принципи побудови стегосистем:

- криптоаналітик має повне подання про стеганографічну систему й деталі її реалізації.

- єдиною інформацією, що залишається невідомою криптоаналітику, є ключ, за допомогою якого тільки його власник може встановити факт присутності й зміст схованого повідомлення; якщо криптоаналітик якимось чином довідається про факт існування схованого повідомлення, це не повинне дозволити йому розшифрувати подібні повідомлення в інших даних доти, поки ключ зберігається в таємниці;

- криптоаналітик повинен бути позбавлений яких-небудь технічних й інших переваг у розпізнаванні або розкритті змісту таємних повідомлень.

- у якості даних може використовуватися будь-яка інформація: текстове повідомлення, зображення, відео- або аудіо-файл, і т.п. Далі будемо використати слово "повідомлення", тому що повідомленням може бути як текст або зображення, так й, наприклад, аудіодані. Для позначення прихованої інформації, будемо використовувати саме термін повідомлення.

Основні терміни й визначення.

Стеганографічна система або стегосистема - сукупність засобів і методів, які використовуються для формування прихованого каналу передачі інформації.

Контейнер - будь-яка інформація, призначена для приховання таємних повідомлень; порожній контейнер - контейнер без убудованого повідомлення; заповнений контейнер або стего - контейнер,що містить вбудовану інформацію.

Вбудоване (приховане) повідомлення - повідомлення, що вбудоване в контейнер.

Стеганографічний канал або просто стегоканал - канал передачі.

Стегоключ (просто ключ) - секретний ключ, необхідний для приховування інформації. Залежно від кількості рівнів захисту (наприклад, вбудовування попередньо зашифрованого повідомлення) у стегосистемі може бути один або декілька стегоключів. Як і у криптографії, за типом стегоключа стегосистеми можна розділити на два типи: із секретним ключем; з відкритим ключем. У стегосистемі із секретним ключем використовується один ключ, що повинен бути визначений або до початку обміну секретними повідомленнями, або переданий по захищеному каналу.

У стегосистемі з відкритим ключем для вбудовування й розшифрування повідомлення використовуються різні ключі, які розрізняються таким чином, що за допомогою обчислень неможливо вивести один ключ із іншого. Тому один ключ (відкритий) може передаватися вільно по незахищеному каналі зв'язку. Крім того, дана схема добре працює й при взаємній недовірі відправника й одержувача. Вимоги до побудови стегосистемы Стегосистема повинна відповідати наступним вимогам: Властивості контейнера повинні бути модифіковані, щоб зміна неможливо була виявити при візуальному контролі. Ця вимога визначає якість приховання впроваджуваного повідомлення: для забезпечення безперешкодного проходження стегоповідомлення по каналу зв'язку воно жодним чином не повинне привернути увагу атакуючого. Стегоповідомлення повинне бути стійким до перекручувань, у тому числі й зловмисних. У процесі передачі зображення (звук або інший контейнер) може набувати різних трансформацій: зменшуватися або збільшуватися, перетворювати в інший формат і т.д. Крім того, воно може бути стиснуто, у тому числі й з використанням алгоритмів стиску із втратою даних. Для збереження цілісності вбудованого повідомлення, необхідне використання коду з виправленням помилки. Можна виділити три тісно зв'язаних між собою й тих, що мають один корінь напрямку стеганографії: - приховання даних (повідомлень), - цифрові водяні знаки -заголовки. Приховання впроваджуваних даних, передача разом з контейнером схованих даних висуває серйозні вимоги до контейнера: розмір контейнера в кілька разів повинен перевищувати розмір даних, що вбудовують. Цифрові водяні знаки використаються для захисту авторських або майнових прав на цифрові зображення, фотографії або інші оцифровані твори мистецтва. Основними вимогами, які пред'являються до таких вбудованих даних, є надійність і стійкість до перекручувань. Цифрові водяні знаки мають невеликий обсяг, однак, з обліком зазначених вище вимог, для їхнього вбудовування використовуються більш складні методи,

ніж для вбудовування просто повідомлень або заголовків. Заголовки використається в основному для маркірування зображень у більших електронних сховищах (бібліотеках) цифрових зображень, аудіо- і відеофайлів. У цьому випадку стеганографічні методи використовуюься не тільки для впровадження ідентифікуючого заголовка, але й інших індивідуальних ознак файлу. Впроваджувані заголовки мають невеликий обсяг, а висунуті до них вимоги - мінімальні: заголовки повинні вносити незначні перекручування й бути стійкі до основних геометричних перетворень.

Обмеження

Залежно від призначення додатків стеганографії пред'являються різні вимоги до співвідношення між стійкістю вбудованого повідомлення до зовнішніх впливів (у тому числі й стегоаналізу) і розміром самого повідомлення, що вбудовується. Для більшості сучасних методів, використовуваних для приховання повідомлення в цифрових контейнерах, має місце зворотна залежність надійності системи від обсягу даних, що приховуються. Дана залежність говорить про те, що при збільшенні обсягу даних, що вбудовують, знижується надійність системи (при незмінності розміру контейнера). Таким чином, використовуваний у стегосистемі контейнер накладає обмеження на розмір даних, що приховуються. Основні принципи комп'ютерної стеганографії й області її застосування

Методи використання надмірності цифрові фотографії, цифрового звуку й цифрового відео

Молодші розряди цифрових відліків містять дуже мало корисної інформації. Їхнє заповнення додатковою інформацією практично не впливає на якість сприйняття, що й дає можливість приховання конфіденційної інформації

Недоліки: За рахунок введення додаткової інформації спотворюються статистичні характеристики цифрових потоків. Для зниження компрометуючих ознак потрібна корекція статистичних характеристик

Переваги: Можливість схованої передачі великого обсягу інформації. Можливість захисту авторського права, схованого зображення товарної марки, реєстраційних номерів і т.п.

Завдання :

1. Проаналізувавши формат BMP, створити програму для організації стегоканалу для приховування даних, що в середньому є в 10 раз меншими за розмір файлу.

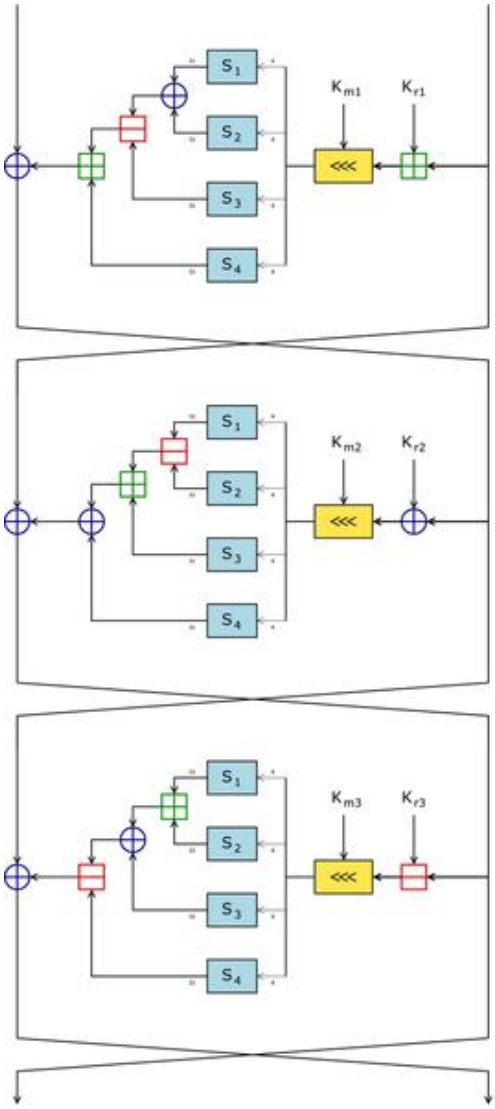
2. Підготувати та захистити звіт.

Лабораторна робота №5

Симетричні блокові шифри на основі мережі Фейстеля

Мета роботи : ознайомитися з методом побудови алгоритмів симетричного блокового шифрування на прикладі мережі Фейстеля.

Теоретичні відомості



Мережа Фе́йстеля (конструкція Фейстеля) — один з методів побудови блокових шифрів. Мережа представляє із себе певну ітеровану структуру, що називається коміркою Фейстеля. При переході від однієї комірки до іншої міняється ключ, причому вибір ключа залежить від конкретного алгоритму. Операції шифрування та дешифрування на кожному етапі дуже прості, і при певній доробці збігаються, вимагаючи тільки зворотного порядку використовуваних ключів. Шифрування за допомогою даної конструкції легко реалізуються як на програмному рівні, так і на апаратному, що забезпечує широкі можливості застосування. Більшість сучасних блокових шифрів використовують мережу Фейстеля як основу. Альтернативою мережі Фейстеля є підстановочно-перестановочна мережа

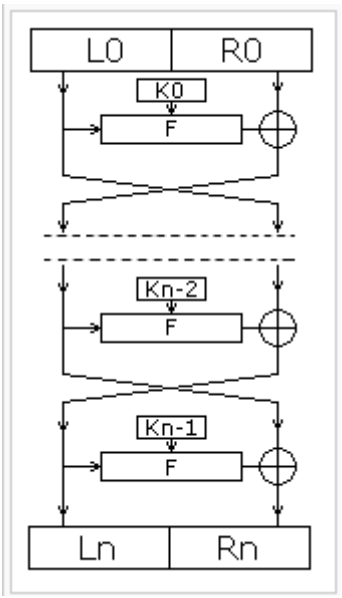
Конструкція блокового шифру на основі мереж Фейстеля

Розглянемо випадок, коли ми хочемо зашифрувати деяку інформацію, представлену у двійковому вигляді в комп'ютерній пам'яті (наприклад, файл) або електроніці, як послідовність нулів й одиниць.

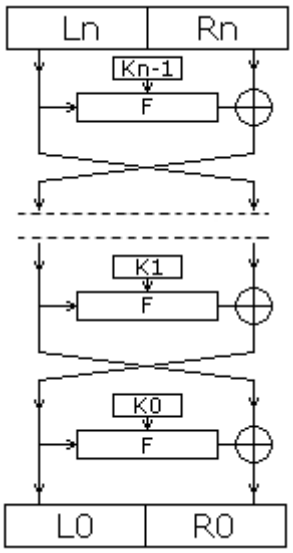
- Вся інформація розбивається на блоки фіксованої довжини. У випадку, якщо довжина вхідного блоку менше, ніж розмір, що шифрується заданим алгоритмом, то блок подовжується яким-небудь способом. Як правило довжина блоку є ступенем двійки, наприклад: 64 біта, 128 біт. Далі будемо розглядати

операції, що відбуваються тільки з одним блоком, тому що з іншими в процесі шифрування виконуються ті ж самі операції.

- Обраний блок ділиться на два рівних подблока — «лівий» (L0) і «правий» (R0).
- «Лівий подблок» L0 видозмінюється функцією f(L0,K0) залежно від раундового ключа K0, після чого він додається по модулі 2 з «правим підблоком» R0.
- Результат додавання присвоюється новому лівому підблоку L1, що буде половиною вхідних даних для наступного раунду, а «лівий подблок» L0 присвоюється без змін новому правому підблоку1 (див. схему), що буде іншою половиною.
- Після чого операція повторюється N-1 раз, при цьому при переході від одного етапу до іншого міняються раундові ключі (K_0 на K_1 і т.д.) за яким-небудь математичним правилом, де N — кількість раундів у заданому алгоритмі. Розшифрування інформації відбувається так само, як і шифрування, з тією лише відмінністю, що ключі йдуть у зворотному порядку, тобто не від першого до N-ному, а від N-го до першого.



Шифрування



Розшифрування

Алгоритмічний опис

1. блок відкритого тексту ділиться на 2 рівні частини $(L_0,\ R_0)$
2. у кожному раунді обчислюється ($i = 1 \ldots n$ - номер раунду)

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

$$R_i = L_{i-1}$$

де f — деяка функція, а Ki − 1 — ключ i-го раунду.

Результатом виконання n раундів є $(L_n,\ R_n)$. Але звичайно в n-ом раунді перестановка Ln й Rn не виконується, що дозволяє використати ту ж процедуру й для розшифрування, просто інвертувавши порядок використання раундової ключової інформації:

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

Невеликою зміною можна домогтися й повної ідентичності процедур шифрування й дешифрування.

Одна з переваг такої моделі — оборотність алгоритму незалежно від використовуваної функції f, і вона може бути як завгодно складною

Математичний опис

Інволюція — взаємо-однозначне перетворення, застосування якого двічі приводить до вихідного значення.

Нехай X — вхідний блок, а A - деяке инволютивное перетворення, Y - вихід.

При однократному застосуванні перетворення: Y = AX, при повторному:

$$AY = A^2X = AAX = X \forall X.$$

Нехай вхідний блок X=(L, R) складається із двох підблоків (L й R) рівної довжини. Визначимо два перетворення $G(X,K) = (L \oplus F(K,R), R)$ (шифрування ключем K) і T(L,R) = (R,L) (перестановка подблоков). Введемо позначення: $\tilde{X} = (\tilde{L}, \tilde{R}) = GX,\ \tilde{\tilde{X}} = (\tilde{\tilde{L}}, \tilde{\tilde{R}}) = G^2X$

Доведемо їх инволютивность:

Нескладно помітити, що перетворення G міняє тільки лівий подблок L, залишаючи правий R незмінним. Тому далі будемо розглядати тільки подблок L. Після того як перетворення G буде двічі застосоване до L одержимо:$\tilde{\tilde{L}} = \tilde{L} \oplus F(K,\tilde{R}) = \tilde{L} \oplus F(K,R) = L \oplus F(K,R) \oplus F(K,R) = L$. У такий спосіб G2X = X, отже G - інволюція.

T2X = T2(L,R) = T(R,L) = (L,R) = X.

Розглянемо сам процес шифрування.

Визначимо X як вхідне значення. Нехай Gi — перетворення із ключем Ki, а Yi — вихідне значення після i-го раунду. Тоді перетворення на i+1-му раунді можна записати у вигляді Yi + 1 = TGiYi, крім першого, де Y1 = TG1X. Отже, вихідне значення після m раундів шифрування буде $Y_m = TG_mY_{m-1} = TG_mTG_{m-1} \dots TG_1X$. Можна помітити, що на останньому етапі не обов'язково виконувати перестановку T.

Розшифрування виконується із застосуванням всіх перетворень у зворотному порядку. У силу інволютивності кожного з перетворень зворотний порядок дає вихідний результат:

$$X = G_1TG_2T \dots G_{m-1}TG_mT(Y_m) = G_1TG_2T \dots G_{m-1}T(Y_{m-1}) = \dots = G_1T(Y_1) = X.$$

Функції, що використовуються у мережі Фейстеля

P-блок (P-box)

Блок перестановок усього лише змінює положення цифр й є лінійним пристроєм. Цей блок може мати дуже велику кількість входів-виходів, однак у силу лінійності систему не можна вважати криптостійкою. Криптоаналіз ключа для n-розрядного P-блоку проводиться шляхом подачі на вхід n-1 різних повідомлень, кожне з яких складається з n-1 нуля («0») і 1 одиниці («1»).

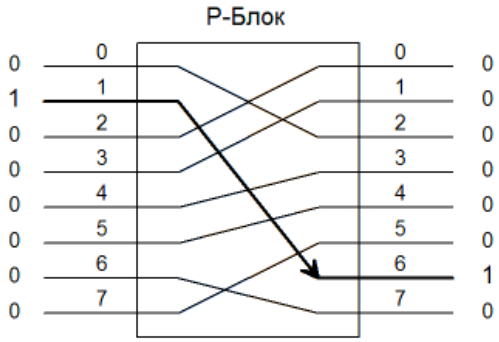


Схема 8-розрядного P-блоку

S-блок (S-box)

Блок підстановок (S-блок) складається з дешифратора, що перетворює n-розрядний двійковий сигнал в однорозрядний сигнал за підстановкою 2n, системи комутаторів внутрішніх з'єднань (усього з'єднань 2n!) і шифраторів, що переводить сигнал з однорозрядного 2n-ого в n-розрядний двійковий. Аналіз n-розрядного S-блоку, при великому n украй складний, однак реалізувати такий блок на практиці дуже складно, тому що число можливих з'єднань украй велике (2n!). На практиці блок підстановок використається як частина більше складних систем.

У загальному випадку S-блок може мати незбіжне число входів/виходів, у цьому випадку в системі комутації від кожного виходу дешифратора може йти не одне з'єднання, а 2 або більше або не йти зовсім. Те ж саме справедливо й для входів шифратора.

В електроніці можна безпосередньо застосовувати наведену праворуч схему, у програмуванні ж генерують таблиці заміни. Обоє цих підходу є еквівалентними, тобто файл, зашифрований на комп'ютері, можна розшифрувати на електронному пристрої й навпаки.

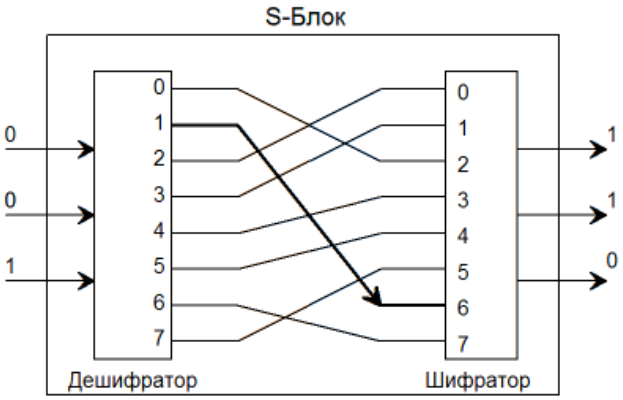
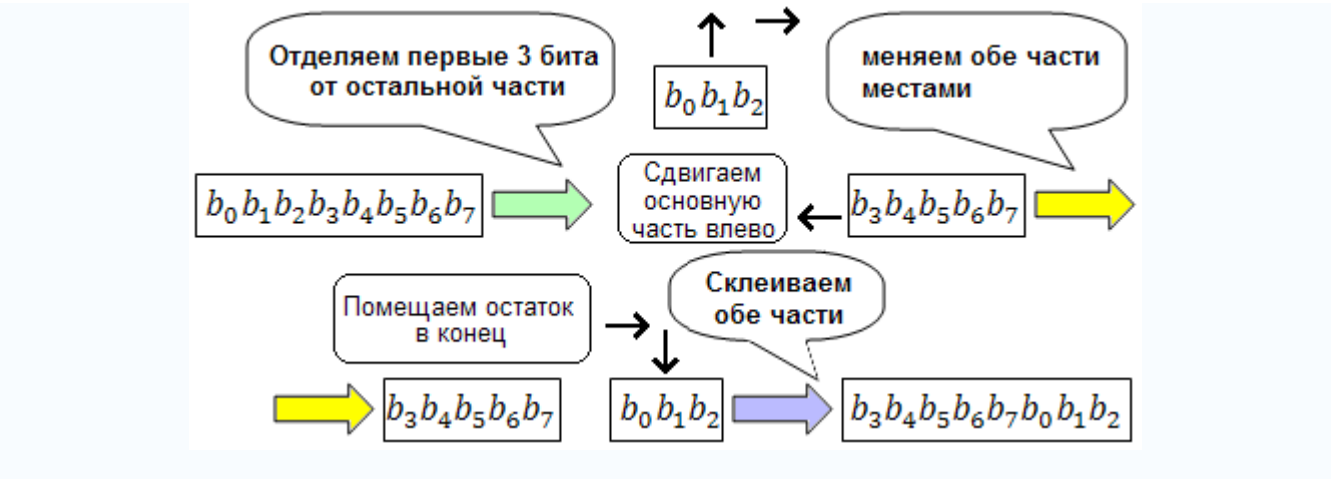


Схема 3-розрядного S-блоку

Таблиця заміни для наведеного розрядн-3-розрядного Блок-блок-S-блоку

№ комбінації	0	1	2	3	4	5	6	7
Вхід	000	001	010	011	100	101	110	111
Вихід	010	110	000	001	011	100	111	101

Циклічний зсув



Циклічний зсув вліво на 3 розряди 8-бітної шини

Можна показати, що циклічний зсув є частиною P-блоку.

У найпростішому випадку (зсув на 1 біт), крайній біт переміщається на інший кінець регістра або шини. Залежно від того який біт береться, правий або лівий, зсув називається вправо або вліво. Зсув на більше число біт можна розглядати, як багаторазове застосування зсуву на 1.

Циклічний зсув на m біт для n-розрядного входу (m < n)		
Напрямок зсуву	Порядок проходження бітів до зсуву	Порядок проходження битов після зсуву
вліво	$b_0,b_1,b_2,...,b_{n-1}$	$b_m,b_{m+1},...b_{n-1},b_0,b_1,...,b_{m-1}$
вправо	$b_0,b_1,b_2,...,b_{n-1}$	$b_{n-m},b_{n-m+1},...b_{n-1},b_0,b_1,...,b_{n-m-1}$

Додавання по модулю n

Операція - (A + B) mod n - це залишок від ділення суми A + B на n, де A й B - числа, що складаються.

Можна показати, що додавання двох чисел по модулі n представляється у двійковій системі числення, як S-блок, у якого на вхід подається число A, а як система комутації S-блоку використається циклічний зсув вліво на B розрядів.

В комп'ютерній техніці й електроніці операція додавання, як правило, реалізована як додавання по модулі n = 2m, де m — ціле (звичайно m дорівнює розрядності машини). Для одержання у двійковій системі A + B mod 2m досить скласти числа, після чого відкинути розряди починаючи з m-того й старше.

Множення по модулю n

Операція - (A * B) mod n — це залишок від ділення добутку A * B на n, де A й B - числа, що перемножуються.

У персональних комп'ютерах на платформі x86 при перемножуванні двох розрядн-m-розрядних чисел виходить число розрядністю 2*m. Щоб одержати залишок від ділення на 2^m потрібно відкинути m старших біт.

Переваги й недоліки

Переваги:

- Простота апаратної реалізації на сучасній елементній базі
- Простота програмної реалізації в силу того, що значна частина функцій підтримується на апаратному рівні в сучасних комп'ютерах (наприклад, додавання по модулю 2 додавання по модулю 2n, множення по модулю 2n, і т.д.)
- Добра вивченість алгоритмів на основі мереж Фейстеля

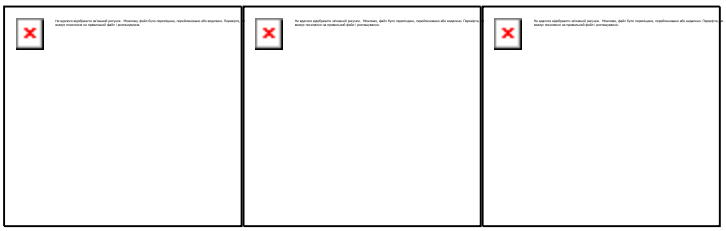
Недоліки:

- За один раунд шифрується тільки половина вхідного блоку

Модифікації мережі Фейстеля

При великому розмірі блоків шифрування (128 біт і більше) реалізація такої мережі Фейстеля на розрядн-32-розрядних архітектурах може викликати ускладнення, тому застосовуються модифіковані варіанти цієї конструкції. Звичайно використаються мережі з 4 розгалуженнями. На рисунку показані найпоширеніші модифікації. Також існують схеми, у яких довжини половинок L0 й R0 не збігаються. Вони називаються незбалансованими.

Модифікації мережі Фейстеля



Тип 1 Тип 2 Тип 3

Приклад реалізації мовою Си

Загальний вид алгоритму шифрування, що використає мережу Фейстеля:

```
/* функція перетворення подблока по ключі (залежить від конкретного алгоритму)
subblock - преутворений подблок
key - ключ
значення, що повертається - перетворений блок*/
int f(int subblock, int key);

/*Шифрування відкритого тексту
left - лівий вхідний подблок
right - правий вхідний подблок
* key - масив ключів (по ключі на раунд)
rounds - кількість раундів*/
void crypt(int *left, int *right, int rounds, int *key)
{
	int i, temp;
	for(i = 0; i < rounds; i++)
	{
		temp = *right ^ f(*left, key[i]);
		*right = *left;
		*left = temp;
	}
}

/*Розшифрування тексту
left - лівий зашифрований підблок
right - правий зашифрований підблок*/
void decrypt(int *left, int *right, int rounds, int *key)
 {
	int i, temp;
	for(i = rounds - 1; i >= 0; i--)
	{
```

```
                temp = *left ^ f(*right, key[i]);
                *left = *right;
                *right = temp;
        }
}
```

Завдання :

1. Створити програму, що реалізує симетричний блоковий алгоритм на основі мережі Фейстеля.
2. Підготувати та захистити звіт.

Література

1. В. Стоуллингс “Криптография и защита сетей – Принципы и практика”, Киев 2003

2. Мельник А.О., Ємець В.Ф., Попович Р. Сучасна криптографія. Основні поняття. Львів, БаК, 2003. – 144 с.

3. Menezes A., van Oorshot P., Vanstone S. Handbook of applied cryptography. CRC Press, 1997

4. Т.Коркішко, А.Мельник , В.Мельник. Алгоритми та процесори симетричного блокового шифрування – Львів, БаК, 2003.

Методичні вказівки до циклу лабораторних робіт з курсу „Захист інформації в комп'ютерних системах" для студентів базового напрямку 6.0915 / Укладачі: В.М. Сокіл, А.А. Андрух, С.О. Власенко – Львів: Національний університет "Львівська політехніка", 2009, 24 с.

Укладачі: В.М. Сокіл, к.т.н., доц. кафедри ЕОМ
А.А. Андрух, асистент кафедри ЕОМ
С.О. Власенко, асистент кафедри ЕОМ

Відповідальний за випуск: В.Т. Кремінь, к.т.н., доц. кафедри ЕОМ

Рецензенти: ВА. Голембо, к.т.н., доц. кафедри ЕОМ.
Ю.В. Морозов, к.т.н., доц. кафедри ЕОМ.

НАВЧАЛЬНЕ ВИДАННЯ

МЕТОДИЧНІ ВКАЗІВКИ
до циклу лабораторних робіт

з дисципліни
"Захист інформації в комп'ютерних системах"

для студентів базового напрямку 6.0915 "Комп'ютерна інженерія "

Укладачі : Сокіл Володимир Михайлович,
Андрух Андрій Анатолійович,
Власенко Сергій Олександрович

Редактор

Комп'ютерне складання

Підписано до друку 200 р.
Формат 70 x 100 $^{1}/_{16}$. Папір офсетний.
Друк на різографі. Умовн. друк. арк. ..... Обл.-вид. арк. .....
Наклад ..... прим. Зам. 780.

Поліграфічний центр
Видавництва Національного університету "Львівська політехніка"
вул. Колесси, 2, 79000, Львів