МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Київський національний університет будівництва і архітектури

ЗАХИСТ ДАНИХ В ІНФОРМАЦІЙНИХ СИСТЕМАХ

Методичні вказівки
до виконання лабораторних робіт
для студентів спеціальностей 015 «Професійна освіта. Комп'ютерні технології», 122 «Комп'ютерні науки», 123 «Комп'ютерна інженерія», 125 «Кібербезпека» та 126 «Інформаційні системи та технології»

Київ 2023

УДК 004.7
338

Укладач: В. М. Вишняков, канд. техн. наук

Рецензент Є.Є. Шабала, канд. техн. наук, доцент

Відповідальний за випуск Ю.І. Хлапонін, д-р техн. наук, професор

Затверджено на засіданні кафедри кібербезпеки та комп'ютерної інженерії, протокол № 2 від 18 вересня 2023 року.

В авторській редакції.

Захист даних в інформаційних системах: Методичні вказівки / 338 уклад.: В.М. Вишняков. – К.: КНУБА, 2023. – 28 с.

Розглянуто методи криптографічного захисту даних та побудови комплексних систем технічного захисту інформації.

Призначено для студентів спеціальностей: 015 «Професійна освіта. Комп’ютерні технології», 122 «Комп’ютерні науки», 123 «Комп’ютерна інженерія», 125 «Кібербезпека» та 126 «Інформаційні системи та технології»

ЗМІСТ

ЗАГАЛЬНІ ПОЛОЖЕННЯ

Лабораторні роботи до курсу “Захист даних в інформаційних системах” призначені для практичного ознайомлення з основами криптографічного захисту комп’ютерної інформації та принципами побудови КСЗІ (Комплексних систем захисту інформації).

Підготовка до кожного заняття та розуміння його мети і змісту – важливі умови набуття стійких практичних навичок.

Кожна лабораторна робота завершується складанням звіту або перевіркою результатів виконання завдань викладачем.

Результат виконання зараховується, якщо студент правильно та охайно склав звіт або надав вірні відповіді на запитання викладача.

У лабораторних роботах курсу “Захист даних в інформаційних системах” використовуються комп’ютери, що підключені до діючої комп’ютерної мережі. Це вимагає від студентів відповідального ставлення до власних дій. Особливо це стосується тих команд і режимів, які дозволяють змінювати параметри налаштування діючої комп’ютерної мережі.

Порядок оформлення звіту до лабораторних робіт

Звіт виконується на аркушах формату А4 та має містити такі частини: титульний аркуш з номером та назвою лабораторної роботи, кодом групи та прізвищем студента; план роботи; мережеві параметри комп’ютера, на якому виконувалась робота; опис дій до кожного пункту плану із конкретними значеннями параметрів, що використовувались у роботі; висновки до результатів по кожному з пунктів плану. Текст звіту повинен бути віддрукований на принтері або відправлений у файлі на електронну адресу викладача.

Оформлення титульного аркушу та рисунків повинні відповідати існуючим стандартам щодо оформлення текстових документів.

ЛАБОРАТОРНА РОБОТА № 1. “Принципи абсолютного захисту інформації”

Мета роботи

Засвоєння принципів абсолютного захисту інформації з використанням шифру Вернама (One-time-pad – одноразовий блокнот).

План роботи

- Підготувати випадкову послідовність бітів у кількості, яка необхідна для шифрування повідомлення з десяти символів.
- Перетворити номер свого телефону у послідовність бітів за допомогою стандартної кодової таблиці UTF-8 або CP866.
- Зашифрувати номер свого телефону з використанням шифру Вернама.
- Розшифрувати номер свого телефону з використанням шифру Вернама.
- Скласти комп'ютерну програму для шифрування шифром Вернама.

Пояснення щодо виконання роботи

Для генерування випадкових (не псевдо випадкових) чисел за допомогою комп'ютера можна скористатись програмою, яку розміщено за адресою:

http://91.198.50.7:11111/exp.html

Результат дії цієї програми показано на рис. 1, а її докладний опис надано у [1]. Значення чисел, які генеруються нею залежать від потоку нейтронів, що впливає на частоту кварцового резонатора комп'ютера, на якому працюємо. Програму можна завантажити на комп'ютер і тоді нею можна користуватись без доступу до мережі. Для цього слід клацнути на "ОК" і за допомогою правої клавіші миші отримати текст коду на мові HTML. Цей текст треба скопіювати у файл типу блокноту і надати йому ім'я exp.html. Клацнувши на файл програма буде запускатись у браузері без звернення до мережі Інтернет.

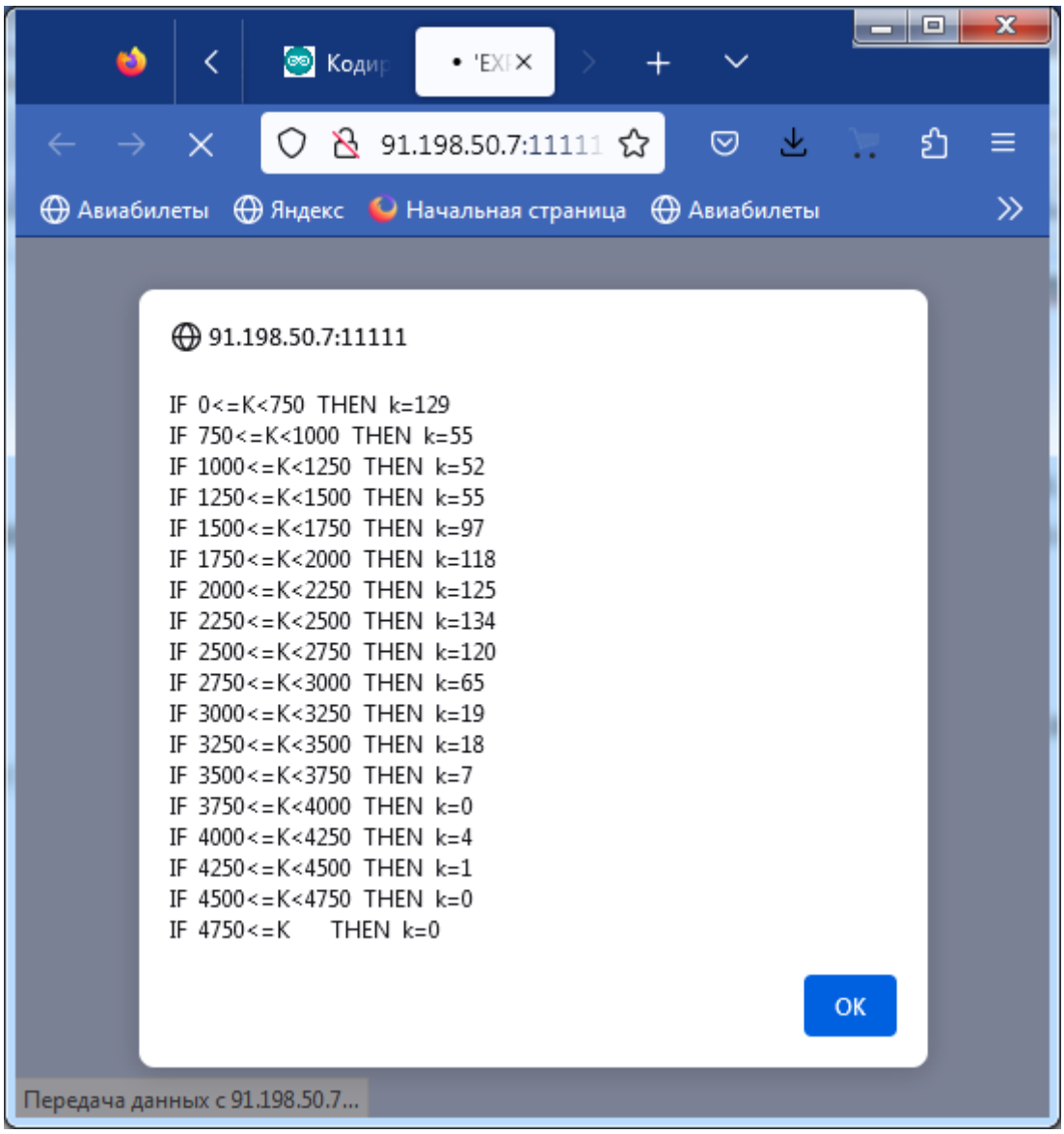


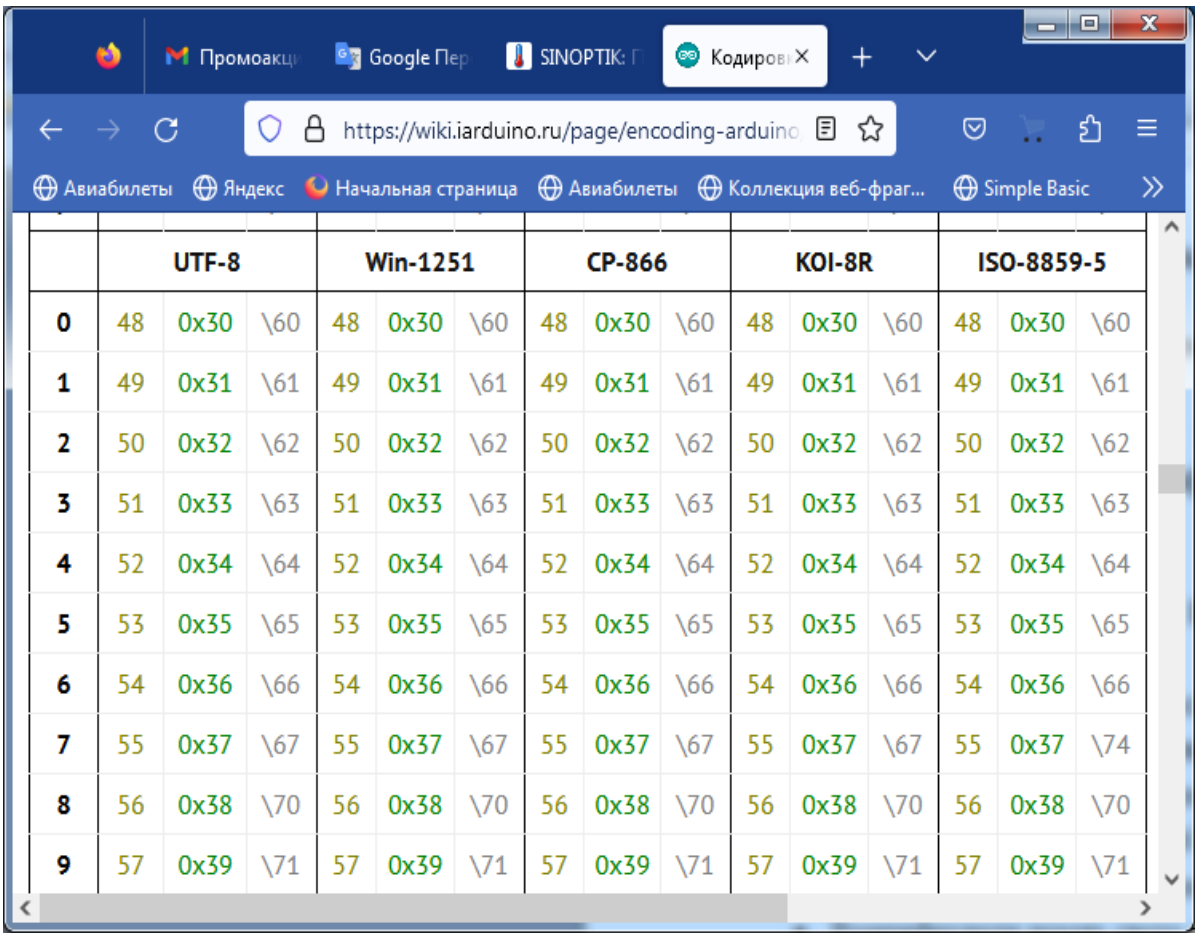
Рис. 1. Результат дії програми exp.html

Щоб отримати потрібну послідовність випадкових бітів за допомогою цієї програми можна обрати низку найбільших значень нарахованої кількості подій, що позначені буквою k у кінці рядків (див. рис. 1). У разі парного k записати 0, а у іншому випадку – 1. З кожного запуску програми будемо отримувати таким чином 8 - 10 випадкових бітів.

Для перетворення символів номеру свого телефону у бітові послідовності можна завантажити кодову таблицю з мережі за адресою:

https://wiki.iarduino.ru/page/encoding-arduino/

Фрагмент цієї таблиці показано на рис. 2.



	UTF-8			Win-1251			CP-866			KOI-8R			ISO-8859-5		
0	48	0x30	\60	48	0x30	\60	48	0x30	\60	48	0x30	\60	48	0x30	\60
1	49	0x31	\61	49	0x31	\61	49	0x31	\61	49	0x31	\61	49	0x31	\61
2	50	0x32	\62	50	0x32	\62	50	0x32	\62	50	0x32	\62	50	0x32	\62
3	51	0x33	\63	51	0x33	\63	51	0x33	\63	51	0x33	\63	51	0x33	\63
4	52	0x34	\64	52	0x34	\64	52	0x34	\64	52	0x34	\64	52	0x34	\64
5	53	0x35	\65	53	0x35	\65	53	0x35	\65	53	0x35	\65	53	0x35	\65
6	54	0x36	\66	54	0x36	\66	54	0x36	\66	54	0x36	\66	54	0x36	\66
7	55	0x37	\67	55	0x37	\67	55	0x37	\67	55	0x37	\67	55	0x37	\74
8	56	0x38	\70	56	0x38	\70	56	0x38	\70	56	0x38	\70	56	0x38	\70
9	57	0x39	\71	57	0x39	\71	57	0x39	\71	57	0x39	\71	57	0x39	\71

Рис. 2. Фрагмент кодової таблиці для кодування цифрових даних за різними стандартами

Як бачимо з цієї кодової таблиці, кодування цифр за різними стандартами нічим не відрізняються.

Для шифрування кожного інформаційного біту шифром Вернама треба скористатись функцією XOR (exclusive disjunction – виключне АБО), результат дії якої показано у табл. 1.

Таблиця 1

Результат дії логічної функції XOR

XOR	A	B
0	0	0
1	0	1
1	1	0
0	1	1

Оформлення результатів лабораторної роботи

Результати кодування та шифрування номеру телефону слід оформити у вигляді таблиць, де для рядків бітів треба обрати шрифт Courier New. Приклад такого оформлення показано у табл. 2 та табл. 3.

Таблиця 2

Шифрування шифром Вернама

Номер телефону	0506497691
Випадкові біти	11010010 11000001 10100000 01000111 10010100
Кодування	00110000 00110101 00110000 00110110 00110100
Шифрування	11100010 11110100 10010000 01110001 10100000

Таблиця 3

Розшифрування шифром Вернама

Шифр	11100010 11110100 10010000 01110001 10100000
Ключ	11010010 11000001 10100000 01000111 10010100
Розшифрування	00110000 00110101 00110000 00110110 00110100
Номер телефону	0506497691

У разі наявності бажання збереження таємниці номеру свого телефону слід змінити одну чи дві будь-які цифри на інші.

У разі наявності знань з програмування складіть комп'ютерну програму для шифрування даних шифром Вернама.

ЛАБОРАТОРНА РОБОТА № 2. “Пошук твірних елементів алгебри груп”

Мета роботи

Практичне знайомство з можливостями використання теорії груп для криптографічного захисту інформації.

План роботи

- Користуючись таблицею варіантів індивідуальних завдань записати у вигляді послідовності цілих чисел усі елементи алгебраїчної групи, яка за розміром відповідає номеру виконавця у списку студентської групи.
- Відшукати твірний елемент знайденої за табл. 4 алгебраїчної групи.
- Описати роль алгебраїчних груп для захисту інформації.

Таблиця 4

Варіанти індивідуальних завдань

№ за списком	Розмір групи	№ за списком	Розмір групи	№ за списком	Розмір групи	№ за списком	Розмір групи
1	13	8	13	15	13	22	13
2	17	9	17	16	17	23	17
3	19	10	19	17	19	24	19
4	23	11	23	18	23	25	23
5	29	12	29	19	29	26	29
6	31	13	31	20	31	27	31
7	37	14	37	21	37	28	37

Пояснення щодо виконання роботи

Розглянемо процедуру виконання роботи на прикладі алгебраїчної групи з 11 елементів. Ця група має наступний вигляд:

0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10.

Оскільки кількість елементів у цій групі є простим числом, то це значить, що у ній повинен бути хоч один твірний елемент. Такий елемент являє собою просте число, яке у різних значеннях степені дає усі елементи грипи, крім нуля. Дії над елементами виконують за правилами арифметики,

але за модулем, який дорівнює числу елементів групи. У даній групі модулем є число 11, яке треба віднімати у разі, якщо результат перевищує значення максимального елемента.

Пошук твірного елемента починаємо з мінімального простого числа 2, для якого послідовно обчислюємо значення степені, починаючи з нульової.

$2^0 = 1$
$2^1 = 2$
$2^2 = 4$
$2^3 = 8$
$2^4 = 5 \pmod{11}$
$2^5 = 10 \pmod{11}$
$2^6 = 9 \pmod{11}$
$2^7 = 7 \pmod{11}$
$2^8 = 3 \pmod{11}$
$2^9 = 6 \pmod{11}$
$2^{10} = 1 \pmod{11}$

Отриманий результат означає, що число 2 є твірним елементом групи, оскільки усі (крім 0) елементи групи отримані.

Інакше треба було б продовжити пошук для чисел 3, 5, 7.

Слід зауважити, що для спрощення обчислень кожне наступне значення степені можна отримувати шляхом множення попереднього на 2.

Наприклад, 2^9 отримуємо як $3*2 = 6$, оскільки $3 = 2^8 \pmod{11}$.

Важливим результатом теорії алгебраїчних груп для захисту інформації є те, що обчислення степені у групах з великою кількістю елементів реалізувати досить легко, а розв'язати зворотну задачу, яку називають дискретним логарифмуванням, майже неможливо. Завдяки цьому було створено криптографічний алгоритм, який використовують майже в усіх сучасних системах захисту даних. Цей алгоритм дозволяє обмінюватись секретними ключами між відправниками та одержувачами інформації з використанням відкритих для прослуховування каналів зв'язку.

ЛАБОРАТОРНА РОБОТА № 3. “Розширення поняття алгебраїчних груп полями Ґалуа”

Мета роботи

Ознайомлення з можливостями криптографічних перетворень над полями Ґалуа.

План роботи

- Ознайомлення з представленням елементів поля Ґалуа для виконання дій додавання та множення.
- Виконання операцій множення за варіантами індивідуальних завдань.
- Ознайомлення з програмою множення елементів поля GF(2^{503}).

Представлення елементів поля Ґалуа

Поля Ґалуа являють собою розширення поняття алгебраїчних груп. Їх позначають GF(p^n), де p – характеристика поля (просте число), n – степінь поля (ціле число), p^n – кількість елементів поля або алгебраїчної групи. У випадку n=1 поле збігається з алгебраїчною групою.

У полях Ґалуа обов'язково є елементи 0 та 1, а також визначені операції додавання, віднімання, множення та ділення. Для полів з характеристикою 2 GF(2^n) результати додавання та віднімання співпадають.

Для поля GF(2^3) на рис. 3 показано три варіанти запису елементів.

0	000	0
1	001	1
2	010	x
3	011	$x+1$
4	100	x^2
5	101	x^2+1
6	110	x^2+x
7	111	x^2+x+1

Рис. 3. Три варіанти запису елементів поля GF(2^3), де у першому стовпчику – десятковий номер, у другому – двійкова форма, а у третьому – поліном

Принцип формування поліному полягає у заміні коефіцієнтів a, b, c на відповідні значення 1 або 0 з двійкової форми у поліномі $ax^2 + bx + c$.

На рис. 4 показано три варіанти запису елементів поля GF(2^4), а на рис. 5 показано степені твірного елементу цього поля.

0	0000	0
1	0001	1
2	0010	x - твірний елемент для усіх GF(2^n)
3	0011	$x+1$
4	0100	x^2
5	0101	x^2+1
6	0110	x^2+x
7	0111	x^2+x+1
8	1000	x^3
9	1001	x^3+1
10	1010	x^3+x
11	1011	x^3+x+1
12	1100	x^3+x^2
13	1101	x^3+x^2+1
14	1110	x^3+x^2+x
15	1111	x^3+x^2+x+1

Рис. 4. Три варіанти запису елементів поля GF(2^4)

$x^0=1$	1	
$x^1=x$	2	
$x^2=x^2$	4	
$x^3=x^3$	8	
$x^4=x+1$	3	- поліном $x^4=x+1$ для заміни степенів, що >3
$x^5=x^2+x$	6	
$x^6=x^3+x^2$	12	
$x^7=x^3+x+1$	11	
$x^8=x^2+1$	5	
$x^9=x^3+x$	10	
$x^{10}=x^2+x+1$	7	
$x^{11}=x^3+x^2+x$	14	
$x^{12}=x^3+x^2+x+1$	15	
$x^{13}=x^3+x^2+1$	13	
$x^{14}=x^3+1$	9	
$x^{15}=1$	1	

Рис. 5. Степені твірного елементу поля GF(2^4)

На рис. 6 показані варіанти запису елементів поля GF(2^5), а також степені твірного елементу цього поля.

0	00000	0	Примітивний поліном: $X^5= X^2+1$					
1	00001	1	$X^0= 1$					1
2	00010	X^1	$X^1= X$				2	
3	00011	$X+1$	$X^2= X^2$					4
4	00100	X^2	$X^3= X^3$					8
5	00101	X^2+1	$X^4= X^4$					16
6	00110	X^2+X	$X^5= X^2+1$					5
7	00111	X^2+X+1	$X^6= X^3+X$					10
8	01000	X^3	$X^7= X^4+X^2$					20
9	01001	X^3+1	$X^8= X^3+X^2+1$					13
10	01010	X^3+X	$X^9= X^4+X^3+X$					26
11	01011	X^3+X+1	$X^{10}= X^4+1$					17_
12	01100	X^3+X^2	$X^{11}= X^2+X+1$				7	
13	01101	X^3+X^2+1	$X^{12}= X^3+X^2+X$					14
14	01110	X^3+X^2+X	$X^{13}= X^4+X^3+X^2$					28
15	01111	X^3+X^2+X+1	$X^{14}= X^4+X^3+X^2+1$					29
16	10000	X^4	$X^{15}= X^4+X^3+X^2+X+1$					31
17	10001	X^4+1	$X^{16}= X^4+X^3+X+1$					27
18	10010	X^4+X	$X^{17}= X^4+X+1$					19
19	10011	X^4+X+1	$X^{18}= X+1$					3
20	10100	X^4+X^2	$X^{19}= X^2+X$					6
21	10101	X^4+X^2+1	$X^{20}= X^3+X^2$					12
22	10110	X^4+X^2+X	$X^{21}= X^4+X^3$				24	
23	10111	X^4+X^2+X+1	$X^{22}= X^4+X^2+1$				21	
24	11000	X^4+X^3	$X^{23}= X^3+X^2+X+1$				15	
25	11001	X^4+X^3+1	$X^{24}= X^4+X^3+X^2+X$				30	
26	11010	X^4+X^3+X	$X^{25}= X^4+X^3+1$					25
27	11011	X^4+X^3+X+1	$X^{26}= X^4+X^2+X+1$					23_
28	11100	$X^4+X^3+X^2$	$X^{27}= X^3+X+1$					11
29	11101	$X^4+X^3+X^2+1$	$X^{28}= X^4+X^2+X$				22	
30	11110	$X^4+X^3+X^2+X$	$X^{29}= X^3+1$				9	
31	11111	$X^4+X^3+X^2+X+1$	$X^{30}= X^4+X$					18
			$X^{31}= 1$					1

Рис. 6. Приклад поля GF(2^5)

Степені твірного елементу у полях Ґалуа дають значення усіх елементів цих полів, крім нульового, що так само, як для алгебраїчних груп чисел, у яких кількість елементів є простим числом. У полях Ґалуа ця кількість може бути не тільки простим числом, а будь яким кратним простому.

Для додавання елементів поля Ґалуа виконують операцію XOR по бітам у двійковій формі. При цьому результат для додавання і віднімання буде однаковий.

Для множення елементів поля Ґалуа використовують поліноми, які їм відповідають. Спочатку знаходять алгебраїчний добуток цих поліномів,

а потім у разі появи членів зі перевищенням степені, яка є максимальною для цього поля, роблять заміну цих членів з використанням примітивного полінома. Для кожного поля Ґалуа існує не менше одного примітивного полінома. Наприклад, для поля $GF(2^4)$ таких поліномів є двоє: $x^4 = x + 1$ та $x^4 = x^2 + 1$.

Приклад множення елементів поля $GF(2^4)$ з використанням полінома $x^4 = x + 1$ має такий вигляд:

$6*7=(x^2 + x)*(x^2 + x + 1)=x^4+\underline{x^3}+\underline{x^2}+\underline{x^3}+\underline{x^2}+x=x^4+x=\underline{x}+1+\underline{x}=1$

Підкреслені доданки знищуються, бо додавання однакових членів дає 0, оскільки додавання співпадає з відніманням.

Варіанти індивідуальних завдань

Таблиця 5

Поля Ґалуа та примітивні поліноми до індивідуальних завдань

№ за списком	$GF(2^4)$ $x^4 = x + 1$	$GF(2^5)$ $x^5 = x^2 + 1$		№ за списком	$GF(2^4)$ $x^4 = x + 1$	$GF(2^5)$ $x^5 = x^2 + 1$
1	3*8	3*18		16	5*8	5*18
2	4*9	4*19		17	6*9	6*19
3	5*10	5*20		18	7*10	7*20
4	6*11	6*21		19	8*11	8*21
5	7*12	7*22		20	9*12	9*22
6	8*13	8*23		21	10*13	10*23
7	9*14	9*24		22	11*14	11*24
8	10*15	10*25		23	12*15	12*25
9	4*8	4*18		24	6*8	6*18
10	5*9	5*19		25	7*9	7*19
11	6*10	6*20		26	8*10	8*20
12	7*11	7*21		27	9*11	9*21
13	8*12	8*22		28	10*12	10*22
14	9*13	9*23		29	11*13	11*23
15	10*14	10*24		30	12*14	12*24

Порядок виконання індивідуального завдання

Користуючись таблицею табл. 5 та рисунками рис. 4 та рис. 6, обрати за своїм номером елементи полів Ґалуа для множення у формі поліномів.

Після знаходження добутків для перевірки знайдених результатів обрати, користуючись рисунками рис. 5 та правою частиною рис. 6, степені твірного елемента, які дорівнюють варіантам індивідуального завдання.

Наприклад, для добутку елементів 6 та 7, що дорівнює 1 для поля $GF(2^4)$, це будуть степені x^5 та x^{10}. Виконати операцію додавання показників степені, а саме $x^5 * x^{10} = x^{5+10} = x^{15}$. Впевнитись, що x^{15} також дорівнює 1.

У разі якщо сума показників степені перевищує 15 для поля $GF(2^4)$, слід відняти 15. Для поля $GF(2^5)$, де максимальне значення степені дорівнює 31, слід у подібних випадках віднімати 31.

Програма множення елементів поля $GF(2^{503})$

У цій програмі використано три масиви з 503 бітів. M1[i], M2[j] та R[i], у які заносять біти множників та біти результату, відповідно. Нульові елементи цих масивів не використовуються. Примітивний поліномом $x^{503} = x^3 + 1$ обрано для заміни степенів, що перевищують 502.

```
function MULT()
{ var i,j,r,r1,r2,r3; for (i=1;i<=503;i++) R[i]=0;
      // Спочатку занесли нулі у масив бітів результату
 for (i=1;i<=503;i++) // Початок циклу множення
       if (M1[i]==1) // Обираємо лише одиниці, бо множеня на 0 дає 0
       {for (j=1; j<=503; j++)
       if (M2[j]==1) // Обираємо лише одиниці другого множника
       {r=i+j-1; // Степінь результату є сумою степенів множників
             if (r>503) // У разі перевищення степені результату
             {r=r-503; // віднімаємо 503
                   if (r>=501)  // У разі другого перевищення
                   {r=r-501;  r1=1+r;  r2=4+r;  r3=501+r;
                   if (R[r3]==0) R[r3]=1; else R[r3]=0;
                   } // Додавання у разі другого перевищення
                   else {r1=r; r2=r+3;}
             if (R[r1]==0) R[r1]=1; else R[r1]=0;
             if (R[r2]==0) R[r2]=1; else R[r2]=0;
             } // Додавання за модулем 2 у разі перевищень
             else {if (R[r]==0) R[r]=1; else R[r]=0;}
       }  // Додавання за модулем 2 у разі відсутності перевищень
 }
} // End of function MULT()
```

ЛАБОРАТОРНА РОБОТА № 4. “Обмін ключами за алгоритмом Діффі-Геллмана”

Мета роботи

Практичне знайомство з алгоритмом Діффі-Геллмана (*Diffie–Hellman key exchange*) за умов використання полів Галуа.

План роботи

- Ознайомлення з методом піднесення до степеня (exponentiation) над елементами полів Галуа.
- Виконання обміну ключами за варіантами індивідуальних завдань.
- Ознайомлення з програмою піднесення до степеня елементів поля $GF(2^{503})$.

Роз'яснення щодо виконання лабораторної роботи

Складність реалізації процедури піднесення до степені елементів поля Галуа шляхом багатократного множення елемента самого на себе буде зростати лінійно зі збільшенням кількості елементів поля. Це є неприпустимим для полів з великою кількістю елементів, але саме такі поля необхідні для досконалого захисту інформації. Тому для обчислення будь якого значення степені заданого елементу A поля $GF(2^n)$ спочатку створюють масив степенів цього елементу, який виглядає так:

$$MA = (A^1, A^2, A^4, A^8, A^{16}, A^{32}, \dots, A^N),$$

де $N = 2^{n-1}$.

Кількість елементів у цьому масиві дорівнює n. Оскільки будь-який показник степеню у полі $GF(2^n)$ може бути представлений як сума показників з масиву MA, наприклад: $A^{372}=A^{256+64+32+16+4}=A^{256}*A^{64}*A^{32}*A^{16}*A^4$. У цьому прикладі замість 371 операції множення достатньо виконати лише 4. Для поля $GF(2^{503})$ кількість елементів масиву MA дорівнює 503.

За допомогою описаного методу типовий сучасний комп'ютер обчислює степінь елементів поля $GF(2^{503})$ за декілька секунд, що шляхом багатократного множення елемента потребує витрат часу, які виходять за межі реальності.

Сутність алгоритму Діффі-Геллмана полягає в тому, що маючи відому для усіх алгебраїчну групи або поле Галуа з відомим твірним елементом T (для полів Галуа такий елемент називають примітивним), шляхом

відкритого обміну даними співбесідники отримують спільний таємний ключ. Для цього кожен з них вигадує своє таємне випадкове число. Наприклад, в одного це буде число X, а у другого – число Y. Потім кожен з них обчислює степінь твірного елементу T, а саме T^X – в одного і T^Y – у другого. Далі вони відправляють один одному результат свого обчислення. Отримавши ці результати кожен з них підносить отриманий результат до свого таємного випадкового степеню. Після цього у першого буде отримано ключ T^{YX}, а у другого – ключ T^{XY}. Легко помітити, що ці ключі – однакові.

Порядок виконання лабораторної роботи

Створити випадкові послідовності X та Y по сім бітів для двох умовних співбесідників. Послідовності 0000000 слід відкидати.

Обчислити степені твірного елементу x для обох співбесідників, а саме x^X та x^Y. Масив MA, який потрібен для цих розрахунків надано на рис. 6.

$x^1 = x$	**2**	**0000010**
$x^2 = x^2$	**4**	**0000100**
$x^4 = x^4$	**16**	**0010000**
$x^8 = x^2 + x$	**6**	**0000110**
$x^{16} = x^4 + x^2$	**20**	**0010100**
$x^{32} = x^4 + x^2 + x$	**22**	**0010110**
$x^{64} = x^4 + x$	**18**	**0010010**

Примітивний поліном: $x^7 = x + 1$

Рис. 6. Значення елементів масиву MA для твірного елементу поля GF(2^7)

Обчислити ключі x^{YX} та x^{XY} та впевнитись, що вони є однаковими.

Для розрахунку кожного з ключів треба спочатку побудувати масиви MA для елементів x^Y та x^X, відповідно.

Програма піднесення до степеню елементів поля GF(2^{503})

Ця програма підносить елемент A до степеня B за такою формулою:

$$A^B = \prod_{i=1}^{503} A^{B_i}, \tag{1}$$

де $B = \sum_{i=1}^{503} B_i$.

Оскільки будь-яке значення B може бути представлено як сума значень що обираються з ряду 2^0, 2^1, 2^2, 2^3, …, 2^{502}, то для обчислення AB

треба зробити не більше ніж 502 операції множення над елементами поля $GF(2^{503})$.

Ця програма створена з урахуванням можливості паралельного виконання декількох даних процедур на різних ядрах процесора у окремих потоках. Такий підхід дозволяє суттєво прискорити роботу головної програми, бо піднесення до степеня потребує значних витрат процесорного часу. Усі дані обробляються у вигляді символьних рядків або масивів з 503 елементів 1 та 0 (символьних або числових). Нульові елементи масивів не використовуються.

Далі надається текст програми з коментарями.

```
const { workerData, parentPort } = require('worker_threads');
// Визначені параметрів для введення даних та виведення результату
var A = [504]; // Елемент поля для піднесення до степені
var B = [504]; // Показник степені
var MA = new Array(504); // Масив для степенів A (1, 2, 4, 8, 16, ...)
 for(var i=0; i<504; i++) MA[i] = new Array(504); // Доповнення виміру MA
var M1=[504]; // Множник 1 для процедури множення MULT()
var M2=[504]; // Множник 2 для процедури множення MULT()
var R=[504];    // Добуток для процедури множення MULT()
let STR=''; // Символьний рядок для введення та виведення даних
STR=workerData; // Введення значень A та B у символьний рядок
 for (var i=1;i<=503;i++) A[i]=B[i]=0; // Спочатку заносимо нулі
 for (var i=1;i<=503;i++) {if (STR[i-1]=='1') A[i]=1; // Вводимо значення A
 if (STR[i+502]=='1') B[i]=1;} // Вводимо значення B
 for (var i=1;i<=503;i++) MA[1][i] = A[i]; // Заносимо A у перший елемент MA
 for (var I=2;I<=503;I++) // Цикл заповнення решти елементів MA
 { // Кожний наступний елемент масиву буде квадратом попереднього
 for (var J=1;J<=503;J++) M1[J]=M2[J]=MA[I-1][J];
 MULT();
 for (var j=1;j<=503;j++) MA[I][j]=R[j];
 } // Завершено заповнення елементів MA
// Для початку циклу множень заносимо у A значення одиничного елемента
for (var i=1;i<=503;i++) A[i]=0; A[1]=1;
   for (var J=1;J<=503;J++) // Цикл множення елементів MA
   if (B[J]==1) // Обираємо для множення лише одиниці у показнику степені,
  {            // бо множення на нуль дає 0.
  for (var I=1;I<=503;I++) {M1[I]= MA[J][I]; M2[I]=A[I];}
```

```
 // Кожне наступне множення відбувається на результат попереднього,
  MULT(); // а перше – на одиничний елемент поля
  for (var I=1;I<=503;I++) A[I]=R[I]; // Переносимо результат для
  } // наступного множення у циклі, або для виводу у кінці циклу
// Результат піднесення A до степеня B занесено в A
STR=''; // Спорожнили символьний рядок для виводу результату
for (var i=1;i<=503;i++) {if (A[i]==1) STR=STR+'1'; else STR=STR+'0';}
// Перенесли результат піднесення A до степеня B у рядок STR
parentPort.postMessage(STR); // Відправили результат у головну програму
// Функція множення елементів поля Галуа GF(2^503)  поліноміальним
// методом з використанням примітивного поліному [x^503=x^3+1
function MULT()
{ var i,j,r,r1,r2,r3; for (i=1;i<=503;i++) R[i]=0;
        // Спочатку занесли нулі у масив бітів результату
         for (i=1;i<=503;i++) // Початок циклу множення
                 if (M1[i]==1) // Обираємо лише одиниці, бо множеня на 0 дає 0
                 {for (j=1; j<=503; j++)
                 if (M2[j]==1) // Обираємо лише одиниці другого множника
                 {r=i+j-1; // Степінь результату є сумою степенів множників
                         if (r>503) // У разі перевищення степені результату
                         {r=r-503; // віднімаємо 503
                                 if (r>=501)  // У разі другого перевищення
                                 {r=r-501;  r1=1+r;  r2=4+r;  r3=501+r;
                                 if (R[r3]==0) R[r3]=1; else R[r3]=0;
                                 } // Додавання у разі другого перевищення
                                 else {r1=r; r2=r+3;}
                         if (R[r1]==0) R[r1]=1; else R[r1]=0;
                         if (R[r2]==0) R[r2]=1; else R[r2]=0;
                         } // Додавання за модулем 2 у разі перевищень
                         else {if (R[r]==0) R[r]=1; else R[r]=0;}
                 }  // Додавання за модулем 2 у разі відсутності перевищень
         }
} // End of function MULT()
```

Спробуйте скласти програму, яка б реалізувала дії за алгоритмом Діффі- Геллмана, що були виконані у цій лабораторній роботі.

ЛАБОРАТОРНА РОБОТА № 5. “Криптографічний захист системи Інтернет голосування”

Мета роботи

Практичне знайомство з засобами захисту від розкриття таємниці голосів та від підробки результатів голосування.

План роботи

- Ознайомлення з технічними засобами захисту конфіденційності голосів виборців та цілісності результатів голосування.
- Ознайомлення з інструкцією щодо експериментального голосування.
- Проведення сеансу експериментального голосування.

Порядок виконання лабораторної роботи

Отримати доступ до системи Інтернет голосування за адресою:

http://vybir.knuba.edu.ua/ або 91.198.50.7 та ознайомитись з можливостями цієї системи через режим "Про дану систему голосування".

Завантажити режим проведення експериментального голосування та за допомогою клавіші "Інструкція для виборців" ознайомитись з інструкцією для проведення експериментального голосування.

Завантажити режим Аудит сервера та отримати перелік файлів системи за допомогою відповідної клавіші у розділі "Аудит файлів адміністратора". У цьому переліку виділити файл CRIPTO.js та скопіювати його назву до режиму Зміст файлу. Отримавши зміст файлу, порівняйте його з Програмою піднесення до степеня елементів поля $GF(2^{503})$, текст якої наведено у лабораторній роботі № 4. Ця програма реалізує усі криптографічні перетворення, які забезпечують захист конфіденційних даних виборців під час зберігання на сервері та під час передавання відкритими каналами мережі Інтернет.

Ознайомтесь з інформацією для виборця у розділі Поради та роз'яснення. Зверніть особливу увагу на останній пункт цих порад, де надані приклади щодо можливості відправки низки безпечних команд на сервер у режимі "Команда ОС". Ці команди дозволяють виборцям впевнитись у штатній роботі сервера та відсутності будь-яких шкідливих втручань в його роботу, у тому числі з боку персоналу, якому доручено керування сервером.

Технологію аудиту докладно описано у роботі [2], а ідею створення подібних систем вперше було розглянуто у роботі [3].

Команди, які слід виконати і задокументувати у цій лабораторній роботі описані у табл. 6.

Таблиця 6

Команди для перевірки роботи сервера

Команда ОС	Що перевіряється
date	Точність серверного годинника
sysctl hw	Конфігурація технічних засобів
ifconfig	Параметри підключення до мережі
ps -aux	Активні процеси серверної ОС

Мета перевірки конфігурації технічних засобів полягає у підтвердженні того, що сервер голосування встановлено на міні комп'ютері (Raspberry Pi 3) з певними технічними обмеженими. Ці обмеження цілком дозволяють провести вибори в межах однієї виборчої дільниці, але не дозволяють імітувати виборчий процес. Перевірка активних процесів дозволяє впевнитись у тому, що ОС після запуску весь час залишалась у робочому стані. Для цього слід порівняти числа зі стовпчику *PID*, що показані на рис. 7, з тими, які зафіксовані у списку активних процесів під час запуску сервера.

```
USER       PID %CPU %MEM   VSZ   RSS TT  STAT    STARTED       TIME COMMAND
root     67427  1.0  0.4  1268  3460 ??  S       7:47PM    0:00.21 sshd: [accep
root     34454  1.7  0.4  1304  3440 ??  S       7:47PM    0:00.37 sshd: kontro
root         1  0.0  0.0   440   316 ??  S      10Feb23    1:32.39 /sbin/init
root     63375  0.0  0.1   724   512 ??  Ip     10Feb23    0:00.18 /sbin/slaacd
_slaacd  51201  0.0  0.1   740   600 ??  Ip     10Feb23    0:00.16 slaacd: fron
_slaacd    584  0.0  0.1   720   576 ??  Ip     10Feb23    0:00.03 slaacd: engi
root     51187  0.0  0.2   776  1984 ??  IpU    10Feb23    0:00.11 syslogd: [pr
_syslogd 83464  0.0  0.1  1376  1316 ??  Sp     10Feb23    3:35.95 /usr/sbin/sy
root     25776  0.0  0.1   720   484 ??  IU     10Feb23    0:00.12 pflogd: [pri
_pflogd  30325  0.0  0.0   760   440 ??  Sp     10Feb23    3:20.47 pflogd: [run
_ntp     76617  0.0  0.3  1152  2384 ??  S<p    10Feb23    0:13.33 ntpd: ntp en
_ntp     30861  0.0  0.2  1052  2248 ??  Ip     10Feb23    0:00.13 ntpd: dns en
root     44087  0.0  0.1  1008  1328 ??  I<pU   11Feb23    0:00.12 /usr/sbin/nt
root     51673  0.0  0.1  1232  1304 ??  S      11Feb23   16:15.10 /usr/sbin/ss
root      3622  0.0  0.2  1988  1964 ??  Ip     11Feb23    0:00.38 /usr/sbin/sm
_smtpd   90982  0.0  0.4  1676  3460 ??  Ip     11Feb23    0:00.16 smtpd: klond
_smtpd   10012  0.0  0.4  1936  3744 ??  Ip     11Feb23    0:00.30 smtpd: contr
_smtpd   11164  0.0  0.4  1772  3684 ??  Ip     11Feb23    0:00.34 smtpd: looku
_smtpd   52004  0.0  0.4  2032  4096 ??  Ip     11Feb23    0:00.59 smtpd: pony
_smtpq   72611  0.0  0.4  2004  3804 ??  Ip     11Feb23    0:00.74 smtpd: queue
_smtpd   26034  0.0  0.4  1664  3528 ??  Ip     11Feb23    0:00.22 smtpd: sched
```

Рис. 6. Результат виконання команди *ps -aux*

Також за допомогою серверу аудиту є можливість перевірити прикладне програмне забезпечення системи голосування, що знаходиться в повному обсязі у файлах адміністратора і повинно бути опубліковане заздалегідь.

Проведіть сеанс експериментального голосування за інструкцією, яка має назву Інструкція для виборців на сервері виборчої дільниці № 999901.

Бажано розподілити ролі між студентами таким чином, щоб на кожного голосуючого припадав аудитор, який перевіряв би коди з'єднань у журналі з'єднань на сервері аудита.

Усі з'єднання виборців із сервером мають унікальні коди, які неможливо підробити. Ці коди захищають виборців від атак посередника. Збіг значення цього коду у журналі, з тим, що отримав виборець, означає відсутність атаки посередника. Форму представлення коду на комп'ютері виборця показано на рис. 7.

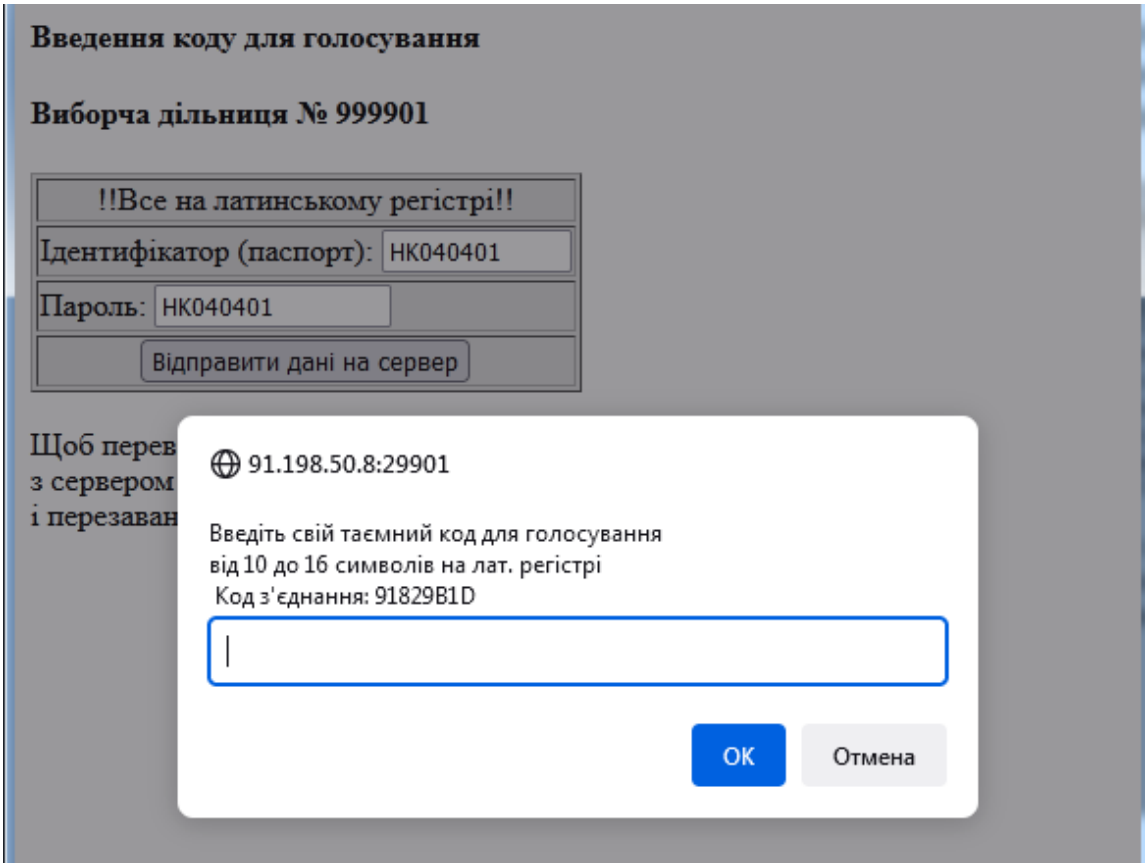


Рис. 7. Представлення коду з'єднання на комп'ютері виборця.

ЛАБОРАТОРНА РОБОТА № 6. “Доповнення системи захисту інформації послугами спостереженості”

Мета роботи

Засвоєння знань щодо вибору послуг захисту інформації за нормативними документами Державної служби спеціального зв'язку та захисту інформації України.

План роботи

- Ознайомлення з документами Державної служби спеціального зв'язку та захисту інформації України.
- Розгляд особливостей щодо вибору тих чи інших послуг захисту під час побудови комплексних систем захисту інформації (КСЗІ).
- Виконання індивідуального завдання щодо доповнення системи захисту послугами спостереженості за обраним варіантом.

Роз'яснення щодо виконання лабораторної роботи

У нормативному документі [5] надано структурну схему критеріїв оцінки захищеності інформації в комп’ютерних системах, яку представлено на рис. 8. У державному стандарті України [6] визначено чотири етапи побудови систем захисту інформації, що є сукупністю методів і засобів забезпечення технічного захисту інформації (ТЗІ). Технічний захист інформації здійснюється поетапно:

1 етап - визначення й аналіз загроз;

2 етап - розроблення системи захисту інформації;

3 етап - реалізація плану захисту інформації;

4 етап - контроль функціонування та керування системою захисту.

Створення КСЗІ починається з розробки моделі загроз, що являє собою опис методів та засобів здійснення кожної з загроз до кожного об’єкту захисту. Щодо кожного об’єкту захисту перш за все слід визначити які саме властивості інформації треба захищати, включаючи цілісність, конфіденційність, а також доступність. Одночасно з цим розробляють модель порушника, де повинні бути описані можливі шляхи реалізації кожної з загроз, а також визначають розмір втрат у грошовому вимірі у разі реалізації кожної з загроз.

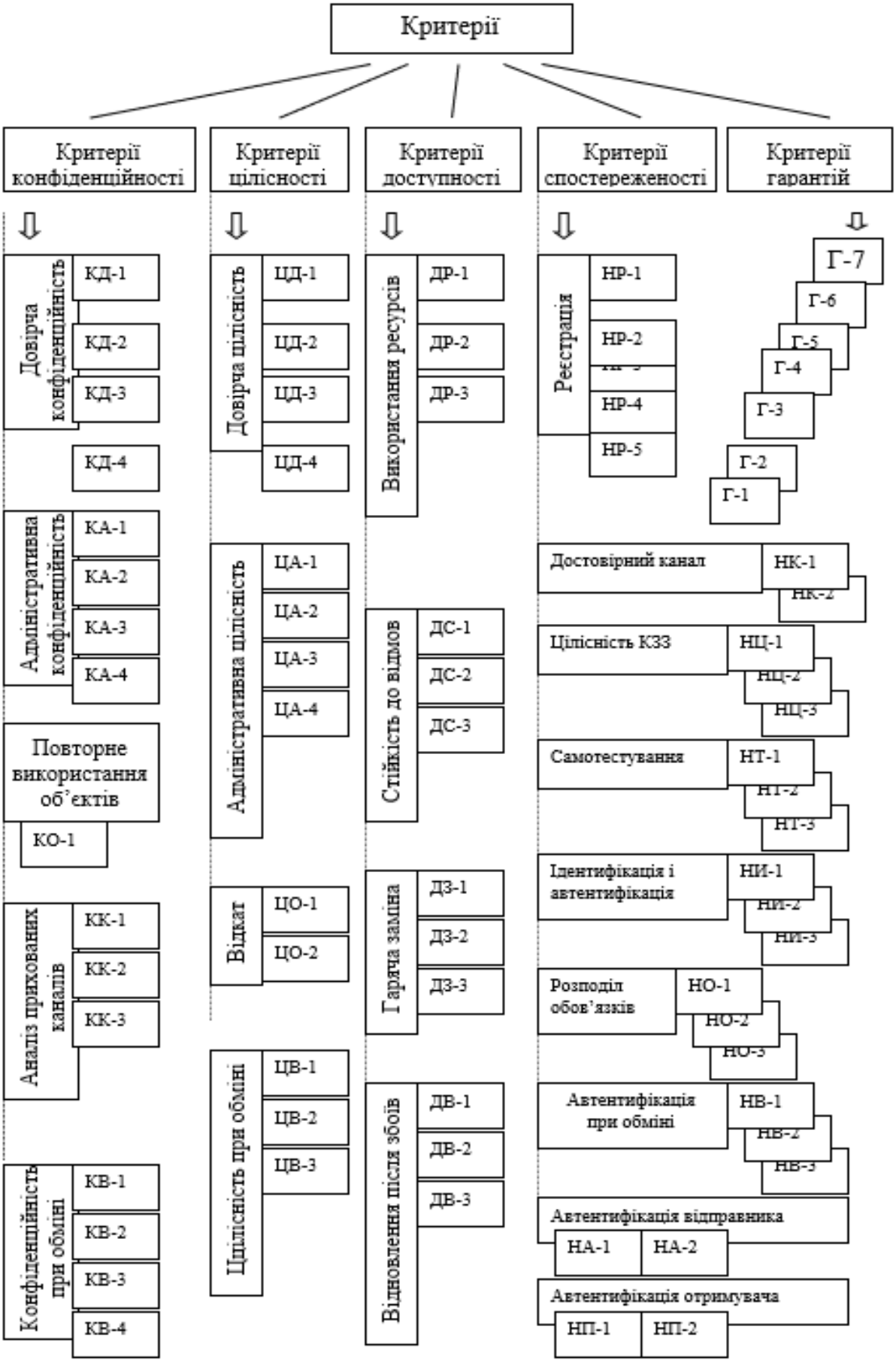


Рис. 8. Загальна структура критеріїв оцінки захищеності інформації

У документі [7] наведено відомості про те як слід захищати інформацію від несанкціонованого доступу, а у документі [8] надано термінологію, яку слід дотримуватись в усіх роботах у цій галузі.

Порядок виконання лабораторної роботи

Обрати варіант індивідуального завдання з табл. 8 за номером у списку навчальної групи.

Зі структура критеріїв оцінки захищеності інформації, яку показано на рис. 8, видалити усе, що не відповідає своєму варіанту.

Доповнити критерії обраного варіанту необхідними умовами з розділів 6, 7, 8 та 9 Нормативного документу НД ТЗІ 2.5-004-99.

Слід звернути увагу, що в даному Нормативному документі в таблицях щодо доповнення функціональних послуг захисту наведено лише необхідні умови, які не завжди є достатніми. Наприклад, на сторінці 4 цього документу вказано: Рівень послуги *цілісність комплексу засобів захисту* (КЗЗ) НЦ-1 є необхідною умовою абсолютно для всіх рівнів всіх інших послуг. При цьому може скластись хибне враження, що завжди слід обирати рівень лише НЦ-1 для усіх критеріїв гарантій. Насправді це не так. Послуга забезпечення цілісності комплексу засобів захисту (КЗЗ) має 3 рівні (див. рис. 8). Тому рівень НЦ-1 слід обирати лише для нижчих рівнів гарантій Г-1 та Г-2. Для середніх рівнів, до яких слід віднести рівні Г-3 та Г-4, треба обирати рівень цілісності КЗЗ НЦ-2, а для високих рівнів гарантій, починаючи з Г-5, слід обирати найвищий рівень цієї послуги НЦ-3. Іншими словами, для обрання тих чи інших рівнів послуг захисту слід орієнтуватись на заданий рівень гарантій.

Для правильного обрання функціональних послуг важливо користуватись Додатком А до Нормативного документу НД ТЗІ 2.5-004-99, де наводиться ряд пояснень з приводу обрання необхідних умов. Наприклад, там вказано, що для рівнів КА-3 і КА-4 необхідною умовою є реалізація рівня КО-1. Додатковою необхідною умовою для всіх рівнів даної послуги є реалізація рівня НО-1, бо в системі повинні бути визначені ролі звичайного користувача і адміністратора. У цьому додатку надано роз'яснення щодо багатьох важливих окремих випадків побудови КЗЗ, без чого можна помилитись у виборі потрібних функціональних послуг захисту у разі цих випадків.

Таблиця 7

Варіанти індивідуальних завдань

№ за списком	Критерій гарантій	Критерії конфіденційності	Критерії цілісності	Критерії доступності
1	Г-1	КД-1, КВ-1	ЦД-1, ЦВ-1	ДР-1, ДС-1, ДЗ-1, ДВ-1
2	Г-2	КД-2, КВ-2	ЦД-2, ЦВ-1	ДР-1, ДС-1, ДЗ-1, ДВ-1
3	Г-3	КД-2, КВ-2	ЦД-2, ЦВ-2	ДР-2, ДС-2, ДЗ-2, ДВ-2
4	Г-4	КД-3, КВ-3	ЦД-3, ЦВ-2	ДР-2, ДС-2, ДЗ-2, ДВ-2
5	Г-5	КД-3, КВ-3	ЦД-3, ЦВ-3	ДР-3, ДС-3, ДЗ-3, ДВ-3
6	Г-6	КД-4, КВ-4	ЦД-4, ЦВ-3	ДР-3, ДС-3, ДЗ-3, ДВ-3
7	Г-7	КД-4, КВ-4	ЦД-4, ЦВ-3	ДР-3, ДС-3, ДЗ-3, ДВ-3
8	Г-1	КД-1, КВ-1	ЦА-1, ЦВ-1	ДР-1, ДС-1, ДЗ-1, ДВ-1
9	Г-2	КД-2, КВ-2	ЦА-2, ЦВ-1	ДР-1, ДС-1, ДЗ-1, ДВ-1
10	Г-3	КД-2, КВ-2	ЦА-2, ЦВ-2	ДР-2, ДС-2, ДЗ-2, ДВ-2
11	Г-4	КД-3, КВ-3	ЦА-3, ЦВ-2	ДР-2, ДС-2, ДЗ-2, ДВ-2
12	Г-5	КД-3, КВ-3	ЦА-3, ЦВ-3	ДР-3, ДС-3, ДЗ-3, ДВ-3
13	Г-6	КД-4, КВ-4	ЦА-4, ЦВ-3	ДР-3, ДС-3, ДЗ-3, ДВ-3
14	Г-7	КД-4, КВ-4	ЦА-4, ЦВ-3	ДР-3, ДС-3, ДЗ-3, ДВ-3
15	Г-1	КА-1, КВ-1	ЦД-1, ЦВ-1	ДР-1, ДС-1, ДЗ-1, ДВ-1
16	Г-2	КА-2, КВ-2	ЦД-2, ЦВ-1	ДР-1, ДС-1, ДЗ-1, ДВ-1
17	Г-3	КА-2, КВ-2	ЦД-2, ЦВ-2	ДР-2, ДС-2, ДЗ-2, ДВ-2
18	Г-4	КА-3, КВ-3	ЦД-3, ЦВ-2	ДР-2, ДС-2, ДЗ-2, ДВ-2
19	Г-5	КА-3, КВ-3	ЦД-3, ЦВ-3	ДР-3, ДС-3, ДЗ-3, ДВ-3
20	Г-6	КА-4, КВ-4	ЦД-4, ЦВ-3	ДР-3, ДС-3, ДЗ-3, ДВ-3
21	Г-7	КА-4, КВ-4	ЦД-4, ЦВ-3	ДР-3, ДС-3, ДЗ-3, ДВ-3
22	Г-1	КА-1, КВ-1	ЦА-1, ЦВ-1	ДР-1, ДС-1, ДЗ-1, ДВ-1
23	Г-2	КА-2, КВ-2	ЦА-2, ЦВ-1	ДР-1, ДС-1, ДЗ-1, ДВ-1
24	Г-3	КА-2, КВ-2	ЦА-2, ЦВ-2	ДР-2, ДС-2, ДЗ-2, ДВ-2
25	Г-4	КА-3, КВ-3	ЦА-3, ЦВ-2	ДР-2, ДС-2, ДЗ-2, ДВ-2
26	Г-5	КА-3, КВ-3	ЦА-3, ЦВ-3	ДР-3, ДС-3, ДЗ-3, ДВ-3
27	Г-6	КА-4, КВ-4	ЦА-4, ЦВ-3	ДР-3, ДС-3, ДЗ-3, ДВ-3
28	Г-7	КА-4, КВ-4	ЦА-4, ЦВ-3	ДР-3, ДС-3, ДЗ-3, ДВ-3

СПИСОК ЛІТЕРАТУРИ

1. Чуприн В.М., Вишняков В.М., Пригара М.П., Генерування випадкових чисел штатними засобами хостів мережі Інтернет, Захист інформації. – 2016. – Т. 18, №4 – С. 323-335. https://jrnl.nau.edu.ua/index.php/ZI/article/view/11085
2. PROOF OF THE POSSIBILITY FOR A PUBLIC AUDIT OF A SECRET INTERNET VOTING SYSTEM Khlaponin, Y., Vyshniakov, V., Komarnytskyi, O. EUREKA, Physics and Engineeringthis link is disabled, 2023, 2023(1), pp. 189–200.
3. Вишняков В.М., Пригара М.П., Воронін О.В. Відкрита система таємного голосування. Управління розвитком складних систем, 2014, №20, С. 110 – 115. http://urss.knuba.edu.ua/files/zbirnyk-20/22.pdf
4. Вишняков В.М. Захист інформації в комп'ютерних системах: Навчальний посібник. – К.: КНУБА, 2022. – 118 с.
5. Д ТЗІ 2.5-004-99. Критерії оцінки захищеності інформації в комп’ютерних системах від несанкціонованого доступу. – Чинний з 28.04.1999. – К.: ДСТСЗІ СБ України, 1999. – 53 с.
6. ДСТУ 3396.0-96. Захист інформації. Технічний захист інформації. Основні положення. – Чинний з 01.01.1997.
7. НД ТЗІ 1.1-002-99. Загальні положення щодо захисту інформації в комп’ютерних системах від несанкціонованого доступу. – Чинний з 28.04.1999. – К.: ДСТСЗІ СБ України, 1999. – 14 с.
8. НД ТЗІ 1.1-003-99. Термінологія в галузі захисту інформації в комп’ютерних системах від несанкціонованого доступу. – Чинний з 28.04.1999. – К.: ДСТСЗІ СБ України, 1999. – 24 с.

Навчально-методичне видання

ЗАХИСТ ДАНИХ В ІНФОРМАЦІЙНИХ СИСТЕМАХ

Методичні вказівки
до виконання лабораторних робіт
для студентів спеціальностей 015 «Професійна освіта. Комп'ютерні технології», 122 «Комп'ютерні науки», 123 «Комп'ютерна інженерія», 125 «Кібербезпека» та 126 «Інформаційні системи та технології»

Укладач: **ВИШНЯКОВ** Володимир Михайлович

Комп'ютерне верстання *М.М. Власенко*

Підписано до друку 18.09.2023 Формат 60 х 84 1/ 16
Ум. друк. арк. 1,63. Обл.-вид. арк. 0,82.
Електронний документ. Вид № 59/ІІІ-17.

Видавець і виготовлювач
Київський національний університет будівництва і архітектури

Повітрофлотський проспект, 31, Київ, Україна, 03680
Свідоцтво про внесення до Державного реєстру суб'єктів
видавничої справи ДК No 808 від 13.02.2002 р.