МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ДВНЗ «УЖГОРОДСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ»
ФАКУЛЬТЕТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ
КАФЕДРА ІНФОРМАТИКИ ТА ФІЗИКО-МАТЕМАТИЧНИХ ДИСЦИПЛІН

МЕТОДИЧНІ ВКАЗІВКИ

для виконання лабораторних робіт

з дисципліни

ТЕХНОЛОГІЇ ЗАХИСТУ ІНФОРМАЦІЇ

для студентів спеціальності

126 «Інформаційні системи та технології»

Ужгород – 2025

УДК 004.056.5:316.774(072)(075.8)
Т 38

Технології захисту інформації: методичні вказівки до виконання практичних робіт для здобувачів першого (бакалаврського) рівня вищої освіти, спеціальності 126 Інформаційні системи та технології факультету інформаційних технологій УжНУ / Укладачі: д.т.н., доц. І.М. Лях, ст. викл. Н.Я. Шумило, асистент П.В. Яворський. Ужгород: УжНУ, 2025. 32 с.

У методичних вказівках до виконання лабораторних робіт з курсу «Технології захисту інформації» сформульовано теми завдань до виконання лабораторних робіт, вимоги до порядку виконання та змісту звіту по проробленій роботі. Також наведена програма навчальної дисципліни та перелік контрольних запитань на підсумковий контроль.

Укладачі: Лях І.М. – д.т.н., доцент, професор кафедри інформатики та фізико-математичних дисциплін факультету інформаційних технологій ДВНЗ «УжНУ»;

Шумило Н.Я. – ст. викладач кафедри інформатики та фізико-математичних дисциплін факультету інформаційних технологій ДВНЗ «УжНУ»;

Яворський П.В. – асистент кафедри інформатики та фізико-математичних дисциплін факультету інформаційних технологій ДВНЗ «УжНУ».

Рецензенти:

д.т.н., проф., завідувач кафедри інформаційних управляючих систем та технологій ДВНЗ «УжНУ» Міца О.В.;

д.т.н., проф., професор кафедри програмного забезпечення систем ДВНЗ «УжНУ» Поліщук В.В.

Рекомендовано кафедрою інформатики та фізико-математичних дисциплін від «27» січня 2025 р., протокол №8.

Розглянуто і схвалено науково-методичною комісією факультету інформаційних технологій УжНУ. Протокол №5 від 28.01.2025 р.

ЗМІСТ

ВСТУП

Актуальність вивчення дисципліни «Технології захисту інформації» сьогодні важко переоцінити через численні загрози, які виникають у сучасному цифровому світі. З розвитком інформаційних технологій і все більшим впливом Інтернету, забезпечення конфіденційності, цілісності та доступності інформації стало однією з основних проблем у різних сферах діяльності: від бізнесу та фінансів до державної безпеки та охорони здоров'я.

Мета вивчення дисципліни **«Технології захисту інформації»** полягає в тому, щоб забезпечити студентів знаннями, навичками та компетенціями, необхідними для ефективного захисту інформаційних систем від різноманітних загроз. Це включає не тільки розуміння теоретичних аспектів захисту даних, а й практичні вміння у застосуванні сучасних технологій для забезпечення безпеки інформації.

Відповідно до освітньої програми, вивчення дисципліни сприяє формуванню у здобувачів вищої освіти таких компетентностей:

ІНТ. Здатність розв’язувати складні спеціалізовані задачі та практичні проблеми в області інформаційних систем та технологій, або в процесі навчання, що характеризуються комплексністю та невизначеністю умов, які потребують застосування теорій та методів інформаційних технологій.

ЗК2.Здатність застосовуватизнанняупрактичнихситуаціях.

ЗК3. Здатність до розуміння предметної області та професійної діяльності.

ФК 6. Здатність використовувати сучасні інформаційні системи та технології (виробничі, підтримки прийняття рішень, інтелектуального аналізу даних та інші), методики й техніки кібербезпеки під час виконання функціональних завдань та обов’язків.

Програма навчальної дисципліни

Змістовий модуль 1.

Тема 1. Захист інформації та його основні завдання.

Тема 2. Поняття про інформацію з обмеженим доступом.

Тема 3. Життєвий цикл розробки систем безпеки.

Тема 4. TCSEC («Оранжева книга») – перший стандарт у галузі оцінки захищеності комп'ютерних систем.

Тема 5. Основні поняття Оранжевої книги.

Тема 6. Common Criteria («Загальні критерії») – європейський стандарт у галузі оцінки захищеності комп'ютерних систем.

Тема 7. НД ТЗІ 2.5-004-99 Критерії оцінки захищеності інформації у комп'ютерних системах від несанкціонованого доступу.

Тема 8. Основні поняття роботи К. Шеннона «Теорія зв'язку в секретних системах».

Змістовий модуль 2.

Тема 9. Симетричні, асиметричні та комбіновані криптосистеми. Їх переваги та недоліки.

Тема 10. Основні вимоги до сучасних криптосистем.

Тема 11. Класифікація сучасних криптосистем та основні вимоги до них.

Тема 12. Математичні основи асиметричної криптографії**.**

Тема 13. DES (Data Encryption Standard) – стандарт шифрування даних США 1977 року.

Тема 14. Основні модифікації DES (3DES, DESX).

Тема 15. Алгоритм криптографічного перетворення ГОСТ 28147-89.

Тема 16. Інші відомі блокові шифри.

Самостійна робота

№ п/п	Назва теми
1.	Захист інформації та його основні завдання.
2.	Поняття про інформацію з обмеженим доступом.
3.	Життєвий цикл розробки систем безпеки.
4.	TCSEC («Оранжева книга») – перший стандарт у галузі оцінки захищеності комп'ютерних систем.
5.	Основні поняття Оранжевої книги.
6.	Common Criteria («Загальні критерії») – європейський стандарт у галузі оцінки захищеності комп'ютерних систем.
7.	НД ТЗІ 2.5-004-99 Критерії оцінки захищеності інформації у комп'ютерних системах від несанкціонованого доступу.
8.	Основні поняття роботи К. Шеннона «Теорія зв'язку в секретних системах».
9.	Симетричні, асиметричні та комбіновані криптосистеми. Їх переваги та недоліки.
10.	Основні вимоги до сучасних криптосистем.
11.	Класифікація сучасних криптосистем та основні вимоги до них.
12.	Математичні основи асиметричної криптографії**.**
13.	DES (Data Encryption Standard) – стандарт шифрування даних США 1977 року.
14.	Основні модифікації DES (3DES, DESX).
15.	Алгоритм криптографічного перетворення ГОСТ 28147-89.
16.	Інші відомі блокові шифри.

Лабораторна робота № 1.

Дослідити рекомендації щодо створення паролів та способи їхнього зламу через атаки підбором, використовуючи словники та правила

Начальна мета: Ознайомити учасників з основними рекомендашими щодо створення надійних паролів, а також з методами їхнього зламу, такими як атаки підбором, використання словників і правил. Розвинути навички забезпечення безпеки особистих даних у цифровому середовищі.

Виховна мета: Формувати у учасників відповідальне ставлення до захисту інформації та інтернет-безпеки. Сприяти розвитку критичного мислення та обережності при створенні паролів і роботі з конфіденційною інформацією.

Завдання:

1. Дослідити основні рекомендації щодо сторення надійних паролів.
2. Ознайомитися з методами злому паролів, такими як атаки підбором, використання словників та правил.

Необхідне обладнання: Індивідуальне робоче місце в комп’ютерному класі, оснащене ПК із встановленою операційною системою, веб-браузером та виділеним місцем для збереження даних.

Короткий теоретичний коментар до теми

Рекомендації щодо створення надійних паролів включають використання довгих і складних паролів, що складаються з поєднання великих і малих літер, цифр і спеціальних символів. Чим довший і складніший пароль, тим важче його зламати. Варто уникати використання простих або очевидних слів, таких як «password», «123456» чи особисті дані, які можна легко дізнатися, як, наприклад, ваше ім'я або дата народження. Крім того, для кожного акаунту краще використовувати унікальний пароль, щоб зловмисники не змогли отримати доступ до всіх ваших акаунтів у разі злому одного з них. Також важливо регулярно змінювати паролі, особливо для важливих акаунтів, і

використовувати двофакторну автентифікацію (2FA) для додаткового захисту. Щоб зберігати складні паролі, можна використовувати менеджери паролів.

Щодо способів зламу паролів, один із найпоширеніших методів — атака методом підбору, коли зловмисники намагаються перебрати всі можливі варіанти комбінацій символів до тих пір, поки не знайдуть правильний. Іншим методом є атака за допомогою словника, коли зловмисники використовують списки найбільш поширених паролів і слів, намагаючись підібрати відповідний. Атака за допомогою правил — це варіант попереднього методу, коли до кожного слова зі словника застосовуються модифікації, наприклад, заміна літер на цифри або додавання спеціальних символів. Також є методи соціальної інженерії та фішинг, де зловмисники намагаються обманним шляхом отримати паролі, створюючи підроблені веб-сторінки чи електронні листи.

Дослідити основні рекомендації щодо сторення надійних паролів

Огляд основних принципів безпеки паролів: Пояснити важливість використання складних паролів для захисту особистої інформації. Дискутувати, чому прості паролі (наприклад "123456" або "password") є ненадійними.

Визначення характеристик надійного пароля: Розглянути, які характеристики має мати надійний пароль — достатня довжина (не менше 12-16 символів), використання різних типів символів (великі й малі літери, цифри, спеціальні символи).

Рекомендації щодо уникнення слабких паролів: Пояснити, чому не варто використовувати особисті дані (ім’я, дата народження, ім’я домашніх тварин) у паролях і як це підвищує вразливість облікових записів.

Використання двофакторної автентифікації (2FA): Ознайомитися з двофакторною автентифікацією як додатковим рівнем захисту для важливих акаунтів. Обговорити її важливість.

Створення та зберігання паролів: Пояснити важливість використання менеджерів паролів для створення і зберігання складних паролів.

Ознайомитися з методами злому паролів, такими як: атаки підбором, використання словників та правил

Атака методом підбору (Brute Force): Пояснити принцип ціст атаки, коли зловмисник намагається перебрати всі можливі комбінації символів, поки не знайде правильний пароль. Описати, чому цей метод ефективний проти слабких паролів, але стас непрактичним для складних і довгих паролів.

Атака за допомогою словника: Розглянути, як зловмисники використовують заздалегідь підготовлені списки популярних паролів, слів і фраз (словники), щоб підібрати пароль. Пояснити, чому важливо уникати використання простих слів у паролях.

Атака за допомогою правил: Ознайомити з техніками, де словники модифікуються за допомогою певних правил (наприклад, заміна бука на цифри, додавання символів). Пояснити, як це може збільшити шанси на успіх атаки навіть на більш складні паролі.

Методи захисту від атак: Розглянути способи захисту від цих атак, такі як збільшення довжини пароля, використання двофакторної автентифікації і застосування сольового шифрування (nashing).

Практичні демонстрації: Провести демонстрацію практичних прикладів цих атак за допомогою спеціального програмного забезпечення чи інструментів для тестування безпеки паролів.

РЕКОМЕНДАЦІЇ ЩОДО ВИКОНАННЯ ПРАКТИЧНОЇ РОБОТИ

1. Перед початком практичної роботи важливо забезпечити безпеку ваших особистих даних. Використовуйте лише тестові акаунти та паролі, щоб уникнути можливих ризиків зловживань реальними даними.
2. Під час виконання роботи застосовуйте на практиці рекомендації щодо створення надійних паролів. Використовуйте комбінації великих і малих літер, цифр і спеціальних символів, забезпечте довжину паролю не менше 12 символів.

3. Використовуйте симулятори атак (метод підбору, словникові атаки, атаки за допомогою правил), щоб проаналізувати, як швидко можна зламати слабкі паролі. Робіть висновки щодо ефективності захисту, застосовуючи різні стратегії.
4. Уникайте використання реальних особистих даних (ім'я, прізвище, дата народження тощо) при створенні паролів під час практичної роботи, щоб уникнути можливих загроз безпеці.
5. Фіксуйте результати роботи: створені паролі, застосовані методи зламу, час, витрачений на злам паролів, та зроблені висновки. Це допоможе вам краще усвідомити важливість створення надійних паролів і оцінити ефективність різних стратегій захисту.
6. Після виконання роботи проаналізуйте слабкі паролі, які ви використовували для тестування, і оцініть, які з них було легко зламати. Це допоможе краще розуміти важливість складності паролів і забезпечити майбутній захист ваших облікових записів.
7. Після завершення практичної роботи обов'язково видаліть всі тестові акаунти та паролі, щоб уникнути їхнього використання сторонніми особами. Перевірте, чи не збереглися ваші паролі в браузері чи інших додатках.
8. В кінці практичної роботи обговоріть з викладачем чи іншими учасниками результати виконаних завдань, помилки, які були допущені, і кроки, які можна вжити для поліпшення безпеки паролів у реальному житті.
9. Оформити звіт у форматі .odf, .doc або .docx.
10. Надіслати файл викладачеві в архіві, що містить звіт.

Лабораторна робота № 2.

Ознайомитись з основними поняттями криптографії та криптоаналізу, вивчити базові алгоритми шифрування та методи атак для зламування зашифрованих даних

Навчальна мета: Ознайомити з основами криптографії та криптоаналізу, розглянути прості алгоритми шифрування і методи атак для розкриття зашифрованих даних.

Виховна мета: Сформувати у учасників розуміння важливості захисту інформації за допомогою криптографії, розвивати відповідальність у використанні методів шифрування та підвищити обізнаність про загрози безпеці даних.

Завдання:

1. Вивчити основні поняття криптографії та криптоаналізу, зокрема базові принципи шифрування та декодування.
2. Ознайомитися з найпростішими алгоритмами шифрування та методами криптоаналізу, а також проаналізувати способи атак для розкриття зашифрованих даних.

Необхідне обладнання: персональне робоче місце в комп'ютерному класі, обладнане ПК із встановленою операційною системою, web-браузером та визначеним місцем для збереження даних.

Короткий теоретичний коментар до теми

Криптографія — це наука про методи захисту інформації, що включає в себе процеси шифрування та дешифрування даних. Її основною метою є забезпечення конфіденційності, цілісності та автентичності переданої інформації. Криптографічні методи використовуються для захисту даних у різних сферах, включаючи банківські транзакції, передачу особистих даних через інтернет, а також у військових та урядових установах.

Шифрування здійснюється за допомогою алгоритмів, які перетворюють зрозуміле повідомлення на зашифровану форму, що неможливо прочитати без ключа для дешифрування. Існують різні типи шифрів, наприклад, симетричне (де один ключ використовується для шифрування та дешифрування) і асиметричне шифрування (де використовуються пари ключів — публічний для шифрування і приватний для дешифрування).

Криптоаналіз, в свою чергу, є наукою, що вивчає методи злому шифрів та розкриття зашифрованих даних без знання ключа. Серед основних методів криптоаналізу — атаки підбором (brute force), атаки на основі частотного аналізу (особливо для простих шифрів, таких як шифр Цезаря) і більш складні методи для зламу сучасних криптографічних алгоритмів.

Розуміння основ криптографії та криптоаналізу є важливим для оцінки рівня безпеки інформації та розробки ефективних методів захисту даних.

Вивчити основні поняття криптографії та криптоаналізу, зокрема базові принципи шифрування та декодування

Ознайомлення з криптографією: Розібратися в основах криптографії, її значенні та завданнях. Пояснити, що таке криптографія, її історію розвитку та важливість у сучасному світі. Акцентувати увагу на ролі криптографії у забезпеченні конфіденційності, цілісності та автентичності інформації.

Принципи шифрування: Вивчити основні методи шифрування, такі як симетричне та асиметричне шифрування. Ознайомитися з поняттями ключа шифрування, відкритого та закритого ключів. Пояснити, як працюють різні алгоритми шифрування (наприклад, DES, AES, RSA) і їхні переваги та недоліки.

Декодування (дешифрування): Розібратися в процесі декодування зашифрованих даних за допомогою відповідного ключа. Пояснити, як алгоритми дешифрування працюють і чому важливо зберігати ключі в безпеці.

Криптографічні функції: Ознайомитися з важливими криптографічними функціями, такими як хешування, цифрові підписи та перевірка автентичності.

Завдання 2. Ознайомитися з найпростішими алгоритмами шифрування та методами криптоаналізу, а також проаналізувати способи атак для розкриття зашифрованих даних

Прості алгоритми шифрування: Вивчити найпростіші алгоритми шифрування, такі як шифр Цезаря, шифр Віженера, та їхні основи. Ознайомитися з принципами роботи цих алгоритмів і застосуванням для шифрування текстів, а також з їхніми слабкими сторонами.

Методи криптоаналізу: Ознайомитися з основними методами криптоаналізу, зокрема:

Атака методом підбору (Brute Force): Оцінити ефективність цього методу для злому простих шифрів, коли зловмисник намагається перебрати всі можливі варіанти ключів.

Частотний аналіз: Розібратися, як використовувати частотний аналіз для зламу простих шифрів (наприклад, шифр Цезаря), вивчаючи частоту появи літер у зашифрованому тексті.

Атака на основі статистичних закономірностей: Ознайомитися з методами пошуку закономірностей у зашифрованих повідомленнях, що дозволяють здогадатися про ключі або шаблони шифрування.

Аналіз способів атак на сучасні криптографічні алгоритми: Розглянути більш складні атаки, такі як атака на основі часу, атака на основі вибору відкритих текстів (chosen plaintext) або атака на основі вибору зашифрованих текстів (chosen ciphertext). Зрозуміти, чому сучасні алгоритми шифрування, такі як AES або RSA, мають високу стійкість до таких атак.

Захист від атак: Ознайомитися з сучасними методами захисту криптографічних алгоритмів від атак. Обговорити важливість використання довгих ключів, регулярної зміни ключів та застосування додаткових технологій, таких як двофакторна автентифікація (2FA) і протоколи безпечного зв'язку (TLS/SSL).

РЕКОМЕНДАЦІЇ ЩОДО ВИКОНАННЯ ПРАКТИЧНОЇ РОБОТИ

1. Ознайомлення з теоретичним матеріалом:
   - Вивчіть основні поняття криптографії та криптоаналізу.
   - Ознайомтесь з базовими методами шифрування (симетричне та асиметричне шифрування).
   - Ознайомтесь з основними методами криптоаналізу: атаки підбором, частотний аналіз, атаки на основі вибору відкритого тексту.
2. Виконання практичних завдань:
   - Проведіть шифрування тексту за допомогою простих алгоритмів (наприклад, шифр Цезаря або Віженера).
   - Використовуйте методи криптоаналізу для розшифровки зашифрованого тексту, застосовуючи метод підбору або частотний аналіз.
3. Аналіз результатів:
   - Оцініть ефективність різних методів шифрування та криптоаналізу.
   - Порівняйте прості алгоритми шифрування з більш складними методами захисту даних.
4. Оформити звіт.

Вимоги щодо оформлення та порядку подання звіту практичної роботи

1. У звіті до цієї роботи повинні бути зазначені:

- Номер практичної роботи, прізвище та ініціали студента, шифр навчальної групи; мета роботи;
- Результати виконання завдань № 1,2;
- Відповіді на запитання (пункт «Порядок і рекомендації щодо виконання роботи»).

2. Звіт має бути оформлений в електронному вигляді у форматах *.odf, *.doc або *.docx.

3. Надіслати викладачу лист з архівом (файл назвати ТЗІ_Lab_02_Прізвище_Ініціали*), який містить файл зі звітом.

Лабораторна робота № 3.

Дослідити алгоритми шифрування DES та 3DES, вивчити режими їх роботи та реалізувати ці алгоритми програмно

Навчальна мета: Ознайомити студентів з алгоритмами шифрування DES та 3DES, розкрити їх принципи роботи, режими шифрування та здійснити програмну реалізацію цих алгоритмів.

Виховна мета: Сприяти розвитку логічного мислення, уваги до деталей, а також формуванню навичок самостійного розв'язування технічних задач у сфері інформаційної безпеки.

Завдання:

1. Вивчити теоретичні основи алгоритмів шифрування DES та 3DES, а також їх режими роботи.
2. Здійснити програмну реалізацію алгоритмів DES та 3DES з використанням відповідних бібліотек та інструментів.

Необхідне обладнання: Індивідуальне робоче місце у комп'ютерному класі, оснащене ПК із встановленою операційною системою, веб-браузером та визначеним місцем для збереження даних.

Короткий теоретичний коментар до теми

Алгоритми шифрування DES (Data Encryption Standard) та 3DES (Triple DES) є одними з основних методів симетричного шифрування, які використовуються для забезпечення конфіденційності переданої інформації. DES був розроблений у 1970-х роках як стандарт для захисту даних, проте через обмежену довжину ключа (56 біт) з часом став вразливим до атак методом підбору. У відповідь на це був розроблений 3DES, який використовує тричі більше довжину ключа (168 біт), забезпечуючи вищий рівень безпеки.

Основою обох алгоритмів є принцип симетричного шифрування, коли для шифрування та дешифрування використовується один і той самий ключ. У DES і 3DES дані шифруються блоками по 64 біти за допомогою складних математичних

операцій, таких як перестановки та заміни, що робить процес шифрування стійким до прямого аналізу.

Режими роботи алгоритмів, такі як ECB (Electronic Codebook), CBC (Cipher Block Chaining), і інші, визначають, як шифруються блоки даних та як обробляються помилки або однакові блоки даних, що зустрічаються кілька разів у повідомленні.

Незважаючи на те, що 3DES значно покращує безпеку в порівнянні з DES, в даний час він також вважається менш безпечним і повільним порівняно з сучасними алгоритмами, такими як AES (Advanced Encryption Standard). Однак, DES і 3DES все ще використовуються в ряді застарілих систем, і знання цих алгоритмів є важливим для розуміння еволюції криптографії.

Завдання 1. Вивчити теоретичні основи алгоритмів шифрування DES та 3DES, а також їх режими роботи

1. **Ознайомитися з принципами роботи алгоритму DES:**

- Дослідити історію створення та еволюцію DES як стандарту шифрування.
- Зрозуміти основні етапи шифрування та дешифрування, зокрема, використання 56-бітного ключа, розподіл блоку даних на 64 біти та застосування різних математичних операцій (перестановки, заміни, підключення).
- Вивчити структуру блочного шифрування DES, зокрема функцію F (функція дозволу), яка використовує різні пермутації, заміни та XOR операції.

2. Вивчити алгоритм 3DES (Triple DES):

- Зрозуміти причини переходу від DES до 3DES, а саме обмеження безпеки DES.
- Описати принцип роботи 3DES, який здійснює три шифрування DES із трьома різними ключами, що забезпечує значно більший рівень безпеки.
- Розглянути варіанти 3DES із двома або трьома різними ключами та їх вплив на ефективність та безпеку.

3. Режими роботи алгоритмів шифрування:

- Вивчити основні режими роботи DES та 3DES, зокрема: **ECB (Electronic Codebook)** – кожен блок даних шифрується окремо, без зв’язку між блоками.

- **CBC (Cipher Block Chaining)** – блокується зв'язок між блоками шляхом використання попереднього шифрованого блоку.
- **CFB (Cipher Feedback)** та **OFB (Output Feedback)** – режими зворотного зв'язку, які використовуються для потокового шифрування.
- Оцінити переваги та недоліки кожного режиму, враховуючи безпеку та продуктивність.

4. Аналіз безпеки DES та 3DES:

- Оцінити вразливості DES до атак методом підбору та іншим типам атак.
- Порівняти 3DES з іншими алгоритмами шифрування, зокрема з AES, з точки зору безпеки та продуктивності.

Завдання 2. Здійснити програмну реалізацію алгоритмів DES та 3DES з використанням відповідних бібліотек та інструментів

1. Реалізація алгоритму DES:

- Написати програму, яка здійснює шифрування та дешифрування даних за допомогою алгоритму DES.
- Реалізувати основні функції: генерація ключа, блокування даних, використання стандартних операцій (перестановки, заміни, XOR).
- Тестувати роботу програми на різних вхідних даних та порівняти результати з очікуваними.

2. Реалізація алгоритму 3DES:

- Розробити програму для шифрування та дешифрування з використанням 3DES, де здійснюється три шифрування за допомогою трьох різних ключів.
- Реалізувати варіанти 3DES з двома або трьома ключами та провести їх тестування.

3. Імплементація різних режимів роботи:

- Реалізувати декілька режимів роботи алгоритмів DES та 3DES, зокрема ECB, CBC, CFB та OFB.
- Тестувати програми на різних наборах даних, щоб перевірити, як різні режими впливають на результат шифрування та ефективність.

4. Перевірка коректності реалізації:

- Порівняти результати шифрування та дешифрування з очікуваними значеннями для кожного алгоритму та режиму.
- Перевірити правильність виконання дешифрування, забезпечивши відновлення початкових даних після двох операцій (шифрування та дешифрування).

5. Оптимізація та тестування програм:

- Перевірити продуктивність програм на великих обсягах даних.
- Оцінити швидкість роботи та можливі шляхи оптимізації алгоритмів для підвищення ефективності.

РЕКОМЕНДАЦІЇ ЩОДО ВИКОНАННЯ ПРАКТИЧНОЇ РОБОТИ

1. Ознайомлення з теоретичними матеріалами:

- Почніть з вивчення теоретичних основ алгоритмів DES і 3DES. Рекомендується знайти книги, статті та онлайн-ресурси, які детально описують принципи роботи цих алгоритмів, їх історію та розвиток.
- Зосередьтеся на розумінні основних етапів шифрування та дешифрування в DES і 3DES, а також на різних режимах роботи (ECB, CBC, CFB, OFB), їх особливостях та застосуваннях.
- Вивчіть вразливості та обмеження алгоритмів, звертаючи увагу на те, чому 3DES був створений як покращення DES і чому сучасні стандарти криптографії, як AES, стали більш популярними.

2. Розробка програми для шифрування DES та 3DES:

- Після теоретичного ознайомлення розпочніть програмну реалізацію алгоритмів. Використовуйте мову програмування, з якою вам найбільш комфортно працювати (наприклад, Python, C++, Java).
- Реалізуйте шифрування та дешифрування для алгоритму DES, дотримуючись його стандартної структури та принципів.
- Після цього розробіть програму для 3DES, застосовуючи три послідовних проходи DES для шифрування даних. Переконайтеся, що ключі для кожного проходу є різними, і протестуйте коректність результату.

3. Імплементація різних режимів роботи:

- Реалізуйте різні режими роботи для алгоритмів шифрування (ECB, CBC, CFB, OFB). Для кожного режиму налаштуйте відповідний механізм шифрування, щоб забезпечити коректну роботу з різними типами даних.
- Звертайте увагу на особливості кожного режиму: для CBC та інших режимів з ланцюговим зворотним зв'язком необхідно правильно обробляти попередні блоки за допомогою ініціалізаційного вектору (IV).
- Проводьте тестування кожного з режимів для перевірки коректності шифрування і дешифрування.

4. Перевірка коректності та тестування програм:

- Тестуйте ваші програми на різних наборах даних для перевірки коректності шифрування та дешифрування. Переконайтеся, що після дешифрування дані відновлюються в початковому вигляді.
- Оцініть ефективність алгоритмів для великих обсягів даних і порівняйте час виконання для різних режимів шифрування.
- Враховуйте можливі помилки при обробці даних і створюйте механізми для їх обробки та коригування, якщо це необхідно.

5. Оформіть звіт.

Вимоги щодо оформлення та порядку подання звіту практичної роботи

1. У звіті до цієї роботи повинні бути зазначені:

- Номер практичної роботи, прізвище та ініціали студента, шифр навчальної групи; мета роботи;
- Результати виконання завдань № 1,2;
- Відповіді на запитання (пункт «Порядок і рекомендації щодо виконання роботи»).

2. Звіт має бути оформлений в електронному вигляді у форматах *.odf, *.doc або *.docx.

3. Надіслати викладачу лист з архівом (файл назвати ТЗІ_Lab_03_Прізвище_Ініціали*), який містить файл зі звітом.

Лабораторна робота № 4.
Дослідити алгоритм шифрування AES та режими його роботи, а також реалізувати його програмно

Навчальна мета: Ознайомити студентів з алгоритмом шифрування AES, його принципами роботи, режимами шифрування та здійснити програмну реалізацію цього алгоритму.

Виховна мета: Сприяти розвитку навичок критичного мислення, здатності до самостійного аналізу та розв'язання задач у сфері інформаційної безпеки, а також формуванню уваги до безпеки інформаційних систем.

Завдання:

1. Вивчити теоретичні основи алгоритму шифрування AES та його режими роботи (ECB, CBC, CFB, OFB тощо).
2. Здійснити програмну реалізацію алгоритму AES, протестувати його на різних даних та режимах роботи.

Необхідне обладнання: Здійснити програмну реалізацію алгоритму AES, протестувати його на різних даних та режимах роботи.

Короткий теоретичний коментар до теми

Алгоритм шифрування AES (Advanced Encryption Standard) є сучасним стандартом симетричного шифрування, який був розроблений для заміни застарілого DES. AES використовує блоки даних розміром 128 біт і може працювати з ключами довжиною 128, 192 або 256 біт. Це забезпечує високу стійкість до атак, таких як атаки методом підбору, та дозволяє ефективно використовувати його в різних системах, включаючи криптографічні протоколи, безпечні канали зв'язку та захист даних.

Основним принципом роботи AES є виконання послідовності математичних операцій, таких як заміни, перестановки та обчислення зворотних елементів в полі Галуа, що забезпечує криптографічну стійкість алгоритму. AES працює за принципом раундів: для кожного раунду

здійснюються операції заміни, змішування, а також додавання ключа до блоку даних.

Режими роботи алгоритму AES визначають спосіб шифрування великих обсягів даних. Найпоширенішими є:

- **ECB (Electronic Codebook)** – кожен блок даних шифрується окремо, що не забезпечує стійкість до повторень.
- **CBC (Cipher Block Chaining)** – попередній зашифрований блок комбінується з наступним блоком даних перед шифруванням, що робить процес більш стійким до атак.
- **CFB (Cipher Feedback)** і **OFB (Output Feedback)** – дозволяють шифрувати потоки даних, а не блоки, що корисно для потокових протоколів.

AES є одним з найбільш широко використовуваних і надійних алгоритмів шифрування в сучасному світі і є стандартом для захисту даних у багатьох урядових і приватних організаціях.

Завдання 1. Вивчити теоретичні основи алгоритму шифрування AES та його режими роботи (ECB, CBC, CFB, OFB тощо)

1. Ознайомлення з алгоритмом AES:

- Дослідити принципи роботи алгоритму AES, зокрема його структуру, використання блоку даних розміром 128 біт та підтримку трьох різних довжин ключа: 128, 192 і 256 біт.
- Вивчити етапи шифрування AES: ініціалізація, заміни (SubBytes), перестановки (ShiftRows), змішування (MixColumns), додавання ключа (AddRoundKey).
- Зрозуміти, як AES використовує кілька раундів для шифрування даних: 10 раундів для 128-бітного ключа, 12 раундів для 192-бітного ключа і 14 раундів для 256-бітного ключа.

2. Розбір режимів роботи AES:

- **ECB (Electronic Codebook)**: кожен блок даних шифрується окремо, що забезпечує швидкість, але не надає достатньої безпеки для великих обсягів даних, оскільки однакові блоки вхідних даних шифруються однаково.
- **CBC (Cipher Block Chaining)**: забезпечує стійкість до атаки повтором блоків шляхом поєднання кожного блоку з попереднім зашифрованим блоком за допомогою операції XOR. Цей режим зменшує вразливості ECB.
- **CFB (Cipher Feedback)**: дозволяє шифрувати потоки даних, забезпечуючи зворотний зв'язок із попередньо зашифрованим блоком. Це корисно для потокового шифрування даних.
- **OFB (Output Feedback)**: аналогічний CFB, але з меншими проблемами з відновленням помилок, що робить його ефективним для передачі даних через неякісні канали зв'язку.
- Дослідити особливості кожного режиму, їх переваги та недоліки, а також сфери застосування.

3. Аналіз безпеки алгоритму AES:

- Оцінити стійкість AES до сучасних атак, таких як атаки за допомогою перебору ключів (brute-force) та атак за допомогою аналітичних методів.
- Порівняти AES з іншими алгоритмами шифрування, такими як DES і 3DES, в контексті швидкості та стійкості до атак.

Завдання 2. Здійснити програмну реалізацію алгоритму AES, протестувати його на різних даних та режимах роботи

1. Розробка програми для шифрування та дешифрування за допомогою AES:

- Вибрати мову програмування для реалізації алгоритму AES (наприклад, Python, C++, Java).
- Здійснити реалізацію основних етапів шифрування AES: ініціалізація ключа, шифрування кожного блоку за допомогою раундів, застосування операцій заміни, перестановки, змішування та додавання ключа.

- Реалізувати функціонал дешифрування, що використовує зворотні операції для відновлення початкових даних.

2. Імплементація режимів роботи:

- Реалізувати підтримку різних режимів роботи AES, таких як ECB, CBC, CFB та OFB.
- Для режиму **ECB** забезпечити шифрування та дешифрування окремих блоків даних без взаємодії між ними.
- Для режиму **CBC** реалізувати процес ланцюгового зв'язку блоків, використовуючи ініціалізаційний вектор (IV).
- Для режимів **CFB** та **OFB** реалізувати шифрування потоків даних, де вихід попереднього блоку використовується для шифрування наступного блоку.

3. Тестування програми на різних даних:

- Тестувати програму на різних наборах даних (тексти, числа, бінарні дані) для перевірки правильності шифрування та дешифрування.
- Переконатися, що при використанні AES з різними ключами (128, 192, 256 біт) програма працює коректно і відновлює початкові дані після дешифрування.

4. Тестування різних режимів роботи:

- Перевірити роботу програми в різних режимах (ECB, CBC, CFB, OFB) на однакових даних і порівняти ефективність та безпеку цих режимів.
- Зробити порівняння швидкості роботи алгоритму для кожного з режимів на великих обсягах даних.

5. Перевірка на стійкість та ефективність:

- Оцінити стійкість алгоритму до атак за допомогою аналізу його швидкості на великих обсягах даних, а також перевірити можливість пошуку ключа за допомогою методу brute-force.
- Зробити висновки щодо продуктивності та безпеки алгоритму в реальних умовах.

РЕКОМЕНДАЦІЇ ЩОДО ВИКОНАННЯ ПРАКТИЧНОЇ РОБОТИ

1. Ознайомлення з теоретичними основами алгоритму AES та його режимами роботи:’

- Почніть з вивчення теоретичних матеріалів щодо алгоритму AES. Приділіть увагу принципам шифрування, процесам заміни, перестановки, змішування та додавання ключа, а також кількості раундів для різних довжин ключа (128, 192, 256 біт).
- Розгляньте режими роботи AES (ECB, CBC, CFB, OFB), їх переваги та недоліки, а також сфери застосування кожного режиму. Особливу увагу зверніть на різницю між ними в контексті безпеки та ефективності.

2. Розробка програмної реалізації алгоритму AES:

- Виберіть мову програмування, на якій ви будете реалізовувати алгоритм (Python, Java, C++ і т.д.). Рекомендується використовувати бібліотеки, що вже реалізують базові функції AES (наприклад, бібліотеки PyCryptodome для Python).
- Реалізуйте основні етапи алгоритму AES, такі як генерація ключа, шифрування та дешифрування блоку даних, використовуючи різні раунди з операціями заміни, перестановки та додавання ключа.

3. Імплементація режимів роботи AES (ECB, CBC, CFB, OFB):

- Розпочніть з реалізації режиму **ECB** (Electronic Codebook), де кожен блок шифрується окремо.
- Потім переходьте до реалізації **CBC** (Cipher Block Chaining), де необхідно реалізувати механізм ланцюгового зворотного зв’язку і використовувати ініціалізаційний вектор (IV).
- Реалізуйте також режими **CFB** (Cipher Feedback) і **OFB** (Output Feedback) для роботи з потоковими даними. Переконайтеся, що кожен з режимів працює коректно з блоками даних.

4.Тестування та перевірка роботи програми:

- Після реалізації алгоритму та режимів роботи протестуйте програму на різних наборах даних (текст, числа, файли). Перевірте, чи програма правильно шифрує та дешифрує дані.
- Оцініть ефективність роботи програми для різних обсягів даних та різних режимів (EC, CBC, CFB, OFB), звертаючи увагу на швидкість виконання та коректність результатів.
- Проведіть тестування на різних ключах (128, 192, 256 біт) і порівняйте результати. Переконайтесь, що програма відновлює початкові дані після дешифрування.

5. Оформіть звіт.

Вимоги щодо оформлення та порядку подання звіту практичної роботи

1. У звіті до цієї роботи повинні бути зазначені:
2. Номер практичної роботи, прізвище та ініціали студента, шифр навчальної групи; мета роботи;
3. Результати виконання завдань № 1,2;
4. Звіт має бути оформлений в електронному вигляді у форматах *.odf, *.doc або *.docx.
5. Надіслати викладачу лист з архівом (файл назвати ТЗІ_Lab_04_Прізвище_Ініціали*), який містить файл зі звітом.

Лабораторна робота № 5.
Дослідити систему RSA та її режими роботи, а також реалізувати цю систему програмно

Навчальна мета: Ознайомити студентів з теоретичними основами криптографічної системи RSA, її принципами роботи та алгоритмами, що використовуються для шифрування і підпису даних. Розвинути навички програмування в контексті реалізації цієї системи.

Виховна мета: Сприяти розвитку аналітичного мислення та самостійного вирішення задач у сфері інформаційної безпеки, а також формувати навички безпечної роботи з криптографічними методами.

Завдання:

1. Вивчити теоретичні основи криптографічної системи RSA, алгоритми шифрування і дешифрування, а також принципи генерації ключів і їх використання.
2. Здійснити програмну реалізацію системи RSA для шифрування та дешифрування повідомлень, а також для створення і перевірки цифрових підписів.

Необхідне обладнання: індивідуальне робоче місце в комп'ютерному класі, оснащене ПК з встановленою версією операційної системи, web-браузером та визначеним місцем для збереження інформації.

Короткий теоретичний коментар до теми

Криптографічна система RSA є однією з найбільш популярних систем для шифрування та цифрового підпису в сучасній криптографії. Її основна особливість полягає в використанні двох різних ключів: відкритого (для шифрування) і закритого (для дешифрування), що базується на асиметричній криптографії.

Алгоритм RSA ґрунтується на математичній задачі факторизації великих чисел, яка є обчислювально складною. В основі системи лежить обчислення

публічного та приватного ключів на основі двох великих простих чисел. Публічний ключ використовується для шифрування даних або перевірки підписів, а приватний — для дешифрування даних або створення підписів.

Процес генерації ключів у RSA включає:

1. Вибір двох великих простих чисел p і q.

2. Обчислення їхнього добутку $n = p \times q$, що використовується як модуль для обох ключів.

3. Обчислення функції Ейлера для числа n ($\varphi(n)$), яка потрібна для визначення приватного ключа.

4. Вибір відкритого експонента e, який повинен бути взаємно простим з $\varphi(n)$.

5. Обчислення закритого ключа d, який є оберненим числом до еее за модулем $\varphi(n)$.

Основні операції в RSA:

- **Шифрування:** $C = M^e \ mod \ n$, де M — повідомлення, яке перетворюється в числовий формат.
- **Дешифрування:** $M = C^d \ mod \ n$, де C — зашифроване повідомлення.

RSA також широко застосовується для цифрових підписів, де користувач створює підпис на документі, використовуючи свій приватний ключ, а інші користувачі можуть перевірити його за допомогою публічного ключа.

Основною перевагою RSA є її безпека, яка ґрунтується на складності задачі факторизації великих чисел. Однак ефективність цього алгоритму залежить від довжини ключа — чим більший розмір ключа, тим складніше розкрити систему, але і тим більше ресурсів необхідно для її роботи.

RSA є основою багатьох криптографічних протоколів, таких як HTTPS, і є важливим елементом для забезпечення безпеки в інтернет-комунікаціях.

Завдання 1. Вивчити теоретичні основи криптографічної системи RSA, алгоритми шифрування і дешифрування, а також принципи генерації ключів і їх використання

1. Ознайомлення з принципами роботи RSA:

- Вивчити основи асиметричної криптографії, на яких ґрунтується система RSA.
- Дослідити принципи шифрування та дешифрування за допомогою RSA, зокрема роль публічного та приватного ключів.
- Ознайомитися з математичною задачею, на якій базується RSA — факторизацією великих чисел, що є важливим для розуміння стійкості системи.

2. Генерація ключів:

- Вивчити процес генерації публічного і приватного ключів RSA: вибір двох великих простих чисел ppp і qqq, обчислення їхнього добутку n=p×qn = p \times qn=p×q, а також функції Ейлера ϕ(n)\phi(n)ϕ(n).
- Ознайомитися з вибором відкритого експонента eee, який повинен бути взаємно простим з ϕ(n)\phi(n)ϕ(n), та обчисленням приватного ключа ddd, який є оберненим до eee за модулем ϕ(n)\phi(n)ϕ(n).

3. Шифрування та дешифрування:

- Розглянути алгоритм шифрування: як використовується публічний ключ для шифрування повідомлень (формула C=Memod nC = M^e \mod nC=Memodn).
- Дослідити процес дешифрування: як приватний ключ використовується для відновлення початкового повідомлення (формула M=Cdmod nM = C^d \mod nM=Cdmodn).
- Ознайомитися з особливостями шифрування повідомлень у вигляді чисел і можливими способами їх подання.

4. Цифрові підписи:

- Вивчити, як RSA використовується для створення цифрових підписів. Створення підпису відбувається за допомогою приватного ключа, а перевірка підпису — за допомогою публічного.

- Ознайомитися з алгоритмами хешування, які зазвичай використовуються для створення підписів, та як ці підписи застосовуються для підтвердження автентичності даних.

Завдання 2. Здійснити програмну реалізацію системи RSA для шифрування та дешифрування повідомлень, а також для створення і перевірки цифрових підписів

1. Розробка програми для генерації ключів:

- Реалізувати алгоритм генерації ключів RSA. Написати функції для вибору двох великих простих чисел, обчислення nnn, ϕ(n)\phi(n)ϕ(n), вибору відкритого експонента eee та обчислення приватного ключа ddd.
- Створити структуру даних для зберігання публічного та приватного ключа, забезпечивши правильне їх використання в подальших операціях.

2. Реалізація шифрування і дешифрування:

- Реалізувати функції для шифрування та дешифрування повідомлень за допомогою алгоритму RSA.
- Для шифрування перетворюйте повідомлення у вигляді чисел, використовуючи відповідні математичні операції Memod nM^e \mod nMemodn для шифрування та Cdmod nC^d \mod nCdmodn для дешифрування.
- Перевірити коректність роботи цих функцій, шифруючи та дешифруючи прості текстові повідомлення.

3. Реалізація цифрових підписів:

- Реалізувати алгоритм цифрових підписів на основі RSA: створення підпису за допомогою приватного ключа і перевірка підпису за допомогою публічного ключа.
- Вибрати відповідний хеш-алгоритм (наприклад, SHA-256) для генерації підпису на повідомленні. Застосувати цей хеш до повідомлення перед підписуванням.
- Перевірити працездатність створених підписів шляхом їх перевірки з використанням публічного ключа.

4. Тестування реалізації:

- Тестувати систему RSA на різних повідомленнях: шифрувати і дешифрувати текстові дані, перевіряти коректність створених підписів і їх верифікацію.
- Перевірити на практиці роботу системи на невеликих числах і більших даних, оцінюючи швидкість виконання операцій і коректність результатів.
- Випробувати систему RSA з різними параметрами (різні розміри ключів) для порівняння швидкості та ефективності.

РЕКОМЕНДАЦІЇ ЩОДО ВИКОНАННЯ ПРАКТИЧНОЇ РОБОТИ

1. Ознайомлення з теоретичними основами RSA:

- Почніть з вивчення основ криптографії та принципів асиметричного шифрування. Досліджуйте математичні основи алгоритму RSA, зокрема факторизацію великих чисел та принципи генерації ключів.
- Ознайомтесь із процесом шифрування і дешифрування за допомогою публічного та приватного ключів RSA, а також з алгоритмами хешування, які використовуються для цифрових підписів.
- Рекомендується також ознайомитись із прикладами математичних операцій, таких як обчислення модуля та обернених елементів.

2. Розробка програмної реалізації:

- Використовуйте мову програмування, зручну для роботи з великими числами, наприклад Python або C++. Почніть з написання функцій для генерації публічного та приватного ключів RSA, використовуючи два великі прості числа.
- Реалізуйте алгоритм шифрування та дешифрування, використовуючи операції Memod nM^e \mod nMemodn для шифрування і Cdmod nC^d \mod nCdmodn для дешифрування. Переконайтесь, що ваша програма правильно обробляє числа і забезпечує точність операцій.

- Імплементація цифрових підписів:
- Додайте функціональність для цифрових підписів. Реалізуйте алгоритм створення підпису за допомогою приватного ключа та його перевірки за допомогою публічного ключа.
- Використовуйте стандартний хеш-алгоритм, наприклад SHA-256, для створення хешу повідомлення перед підписуванням.
- Перевірте правильність підпису на тестових даних, перевіряючи, чи може система успішно верифікувати підпис за допомогою публічного ключа.
- Тестування і перевірка коректності роботи:
- Тестуйте вашу програму на різних наборах даних (наприклад, прості текстові повідомлення) для перевірки шифрування і дешифрування.
- Переконайтесь, що повідомлення правильно відновлюється після дешифрування. Для перевірки цифрових підписів, протестуйте, чи програма правильно генерує та перевіряє підписи для різних повідомлень.
- Оцініть ефективність алгоритму на великих обсягах даних і різних розмірах ключів (наприклад, 512, 1024, 2048 біт).
- Оформіть звіт.

Вимоги щодо оформлення та порядку подання звіту практичної роботи

1. У звіті до цієї роботи мають бути вказані:
- Номер практичної роботи, прізвище та ініціали студента, шифр навчальної групи, а також мета роботи.
- Результати виконання завдань №1 та №2.
- Відповіді на запитання з пункту «Порядок і рекомендації щодо виконання роботи».
2. Звіт повинен бути оформлений в електронному вигляді у форматах *.odf, *.doc або *.docx.
3. Надіслати викладачу листа з архівом, де файл зі звітом має бути названий як ТЗІ_Pract_05_Прізвище_Ініціали*.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ ТА ДЖЕРЕЛ

1. Інформаційна безпека: теорія та практика / В. С. Ковальов, О. І. Кудрявцев. Київ: НТУУ «КПІ», 2020.

2. Основи криптографії та захисту інформації / В. М. Коваленко, І. В. Дем'яненко. Харків: ХНУРЕ, 2020.

3. Інформаційна безпека в інформаційних системах / О. В. Мартинюк, О. В. Шевчук. Київ: Академвидав, 2021.

4. Безпека комп'ютерних мереж і кіберзахист / Ю. І. Кравець, О. М. Носенко. Київ: Вища школа, 2021.

5. Захист персональних даних в Україні: правові та технічні аспекти / І. М. Бондаренко, І. В. Пилипчук. Київ: ІТФ, 2020.

6. Управління інформаційною безпекою та аналіз ризиків / В. О. Сєркулов, В. М. Іванов. Харків: ХНУРЕ, 2022.

7. Захист від кіберзагроз та кібербезпека / О. Ю. Хрипунов. Київ: Наукова думка, 2021.

8. Системи виявлення вторгнень та їх використання в сучасних інформаційних системах / В. С. Єфімов, О. В. Коваленко. Київ: НТУУ «КПІ», 2022.

9. Основи кібербезпеки та захисту інформації / М. В. Чернявський, С. І. Шевченко. Київ: Літера ЛТД, 2021.

10. Системи захисту інформації: теорія і практика / Ю. С. Яценко, В. М. Соколов. Київ: Діалектика, 2022.

11. Інформаційна безпека в умовах цифрової трансформації» / О. В. Корнієнко, О. М. Колесніков. Київ: Кондор, 2021.

12. Практичні аспекти кібербезпеки / А. С. Романюк. Київ: Наукова думка, 2022.