МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ВСП «ОДЕСЬКИЙ ТЕХНІЧНИЙ ФАХОВИЙ КОЛЕДЖ ОНТУ»

Спеціальність: 123 «Комп'ютерна інженерія»

Освітня програма: «Обслуговування комп'ютерних систем та мереж»

Група: 4КС-56

Дипломний проект

здобувача освіти денної форми навчання
КС.56.06.000.ДП

Дедігурова Микити Олександровича

м. Одеса
2023 р.

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ВСП «ОДЕСЬКИЙ ТЕХНІЧНИЙ ФАХОВИЙ КОЛЕДЖ ОНТУ»

Спеціальність: **123 «Комп'ютерна інженерія»**

Освітня програма: «**Обслуговування комп'ютерних систем та мереж»**

Група: **4КС-56**

ПОЯСНЮВАЛЬНА ЗАПИСКА

до дипломного проекту (роботи) на тему:

Розробка та реалізація алгоритму шифрування e-mail повідомлення

Проектний матеріал складається з пояснювальної записки на 65 сторінках та графічного (презентаційного) матеріалу на 14 аркушах (слайдах).

Дипломник_____(Дедігуров М.О.)

Керівни_____(Шевцов Ю.С.)

Консультанти:

з економічної части_____(Копайгородська Т.Г.)

з охорони праці_____(Чорновол Н.І.)

з дотримання вимог ЄСКД_____(Петрашова В.І.)

старший консультант_____(Кривченко Ю.В.)

До захисту допущений

Голова циклової комісії_____(Кривченко Ю.В.)

Завідувач відділення_____(Скорнякова О.В.)

Захист «21» червня 2023 р. Протокол ДКК № 3

Оцінка ДКК 4 (добре)

Секретар ДКК _____

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ВСП «ОДЕСЬКИЙ ТЕХНІЧНИЙ ФАХОВИЙ КОЛЕДЖ ОНТУ»

Відділення **комп'ютерних систем** Комісія **КТ та ПІ**

Спеціальність **123 «Комп'ютерна інженерія»**

Освітня програма **«Обслуговування комп'ютерних систем та мереж»**

ЗАТВЕРДЖУЮ:

Заст. дир. з НВР Беркань І.В.

"_____" _____ 2023р.

ЗАВДАННЯ

на дипломний проект (роботу)

Здобувачеві (здобувачці) освіти Дедігуров Микита Олександрович

(прізвище, ім'я, по батькові)

1. Тема проекту (роботи): Розробка та реалізація алгоритму шифрування e-mail повідомлень

затверджена наказом по коледжу від "17" **жовтня** 2022 р. № **235-А2-ОД**

2. Термін здачі закінченого проекту (роботи) 12.06.2023

3. Вихідні данні до проекту (роботи): HTML-код. CSS. Adobe Photoshop. WEB-додаток. Ретейлер. Інтернет- магазин. Аналоги. WEB-хостинг. Система управління змістом CMS. UX дизайн. Figma. Доменне ім'я URL- адреса. Маркетинг. SEO та аналітика. Інструменти створення WEB-сайту. B2B2C. Bohemian Coding Sketch. Копірайтинг. Соц.мережі.

4. Зміст розрахунково-пояснювальної записки (перелік питань, які необхідно розробити)

Аналітична частина

Безпека електронної пошти

Алгоритми шифрування

Розробка та реалізація алгоритму шифрування

Економічний розділ

Охорона праці

5. Перелік графічного (презентаційного) матеріалу (з точним зазначенням обов'язкових креслень, кількості слайдів)

Загрози інформаційної безпеки, через користування ресурсами мережі інтернет; Взаємодія протоколівроботи електронної пошти; Приклад симетричного шифрування; Асиметричний метод шифрування ; Принцип дії Шифра Цезаря ; Циліндршифрування тексту алгоритму Скитала; Блок-схема алгоритму шифрування; Реалізація шифрування Цезаря; Результат роботи програми шифрування Цезаря; Реалізація шифру Скитала; Результат роботи програми шифрування Скитала

6. Консультанти по проекту (роботі), із зазначенням розділів проекту, що їх стосується

Розділ	Консультант	Підпис, дата	
		Завдання видав	Завдання прийняв
Вступ, 1	Шевцов Ю.С.		
2	Копайгородська Т.Г.		
3	Чорновол Н.І.		
Нормоконтроль	Петрашова В.І.		

7. Дата видачі завдання _____

Керівник _____
(підпис)

Завдання прийняв до виконання _____
(підпис)

КАЛЕНДАРНИЙ ПЛАН

№з/р	Назва етапів дипломного проекту (роботи)	Термін виконання етапівдипломного проекту (роботи)	Відмітка про виконання
1.	Огляд літератури. Огляд існуючих рішень.	22.05.2023	виконав
2.	Аналыз загроз безпеці електронної пошти та заходів щодо захисту скриньок від онлайн-загроз	24.05.2023	виконав
3.	Огляд методів захисту електронної пошти та протоколів роботи з електронною поштою	29.05.2023	виконав
4.	Огляд симетричного та асиметричного шифрування	01.06.2023	виконав
5.	Анализ гібридного шифрування та шифрування електронної пошти	03.06.2023	виконав
6.	Аналіз алгоритмів шифрування Цезаря та Скитала	05.06.2023	виконав
7.	Шифрування електронної пошти	07.06.2023	виконав
8.	Розробка алгоритму шифрування	09.06.2023	виконав
9.	Підготовка до попереднього захисту, підготовка до захисту	11.06.2023	виконав
10.	Отримання рецензії, відповіді на зауваження рецензента	15.06.2023	виконав
11.	Захист роботи	19.06.2023	виконав

Дипломник _____
(підпис)

Керівник _____
(підпис)

Формат	Зона	Поз.	Позначення	Назва	Кіл.	Примітка
				Документація		
			КС.56.06.000.ДП	Дипломний проект	65	
А4			КС.56.06.000.ДП.ПЗ	Пояснювальна записка	1	

Зм.	Арк.	№ докум.	Підпис	Дата	КС.56.06.000.ДП		
Розробив		Дедігуров М.О.			Розробка та реалізація алгоритму шифрування e-mail повідомлень	Літ.	Аркуш 4 / Аркушів
Перевірив		Шевцов Ю.С.				У Д П	
Н. Контр.		Петрашова В.І.				ОТФК ОНТУ, гр. 4КС-56	
Затверд.		Кривченко Ю.В..					

ЗМІСТ

ВСТУП……..6
1. ТЕХНОЛОГІЧНИЙ РОЗДІЛ…….……..7
1.1 Безпека електронної пошти……..…….7
1.1.1 Загрози безпеці електронної пошти……..7
1.1.2 Заходи щодо захисту скриньок від онлайн-загроз……...9
1.1.3 Огляд методів захисту електронної пошти…….....12
1.1.4 Протоколи роботи з електронною поштою……16
1.2 Алгоритми шифрування……...18
1.2.1 Симетричне шифрування……..19
1.2.2 Асиметричне шифрування……20
1.2.3 Гібридне шифрування……...21
1.3 Розробка та реалізація алгоритму шифрування…….24
1.3.1 Шифрування електронної пошти…….24
1.3.2 Алгоритм шифрування Цезаря…….26
1.3.3 Алгоритм шифрування Скитала……...29
1.3.4 Розробка алгоритму шифрування……30
1.3.5 Шифрування розробленим алгоритмом……..32
1.3.6 Програмна реалізація……38
2 ЕКОНОМІЧНА ЧАСТИНА……45
3 ОХОРОНА ПРАЦІ……..……51
ВИСНОВОКИ…….……56
ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ…….…….57
Додаток А. ……..58
Додаток Б. Слайди мультимедійної презентації…….60

ВСТУП

Електронна пошта – найдешевший і найзручніший спосіб зв'язку в наш динамічний час. Залежно від використовуваних технологій, вона може стати як легко доступною для сторонніх, так і конфіденційною та анонімною.

Коли ми натискаєте кнопку "Надіслати", наш лист спочатку потрапляє на поштовий сервер інтернет-провайдера. Потім, через кілька проміжних комп'ютерів та серверів, воно доставляється до поштової скриньки одержувача. Якщо ми надсилаємо повідомлення у відкритому вигляді, адміністратор провайдера та адміністратори будь-якого з цих проміжних комп'ютерів можуть легко його скопіювати та прочитати. На щастя для користувачів електронну пошту досить легко захистити. Існує безліч безкоштовних чи умовно-безкоштовних програм шифрування. Використання криптографії, один із варіантів підвищення рівня безпеки нашого листування. Якщо шифрувати листи перед відправкою, то противник, який слідкує за вашою електронною поштою, прочитати їх не може. Один із найкращих способів забезпечення анонімності листування заснований на використанні безкоштовних анонімних ремейлерів. Анонімний ремейлер – це комп'ютер, що працює в автономному режимі, який пропускає через себе електронну пошту, приховуючи адреси відправника та одержувача. Якщо об'єднати криптографію та анонімні ремейлери, то можна досягти практично повної конфіденційності та анонімності електронної пошти. Послуги ремейлерів та необхідне програмне забезпечення безкоштовні. На жаль, освоєння цих технологій потребує деяких зусиль та ґрунтовних знань у галузі криптографії та інтернет-технологій.

В інтернеті можна знайти безкоштовні сервіси електронної пошти, які забезпечують шифроване з'єднання. Всі необхідні для роботи зцими сервісами програмні засоби вже включені в операційну систему Windows. При підключенні до сервісу комп'ютер самостійно встановить шифроване з'єднання з комп'ютером компанії.

1. ТЕХНОЛОГІЧНИЙ РОЗДІЛ

1.1 Безпека електронної пошти

1.1.1 Загрози безпеці електронної пошти

Електронна пошта - це один з основних засобів комунікації в інтернеті, але разом з тим вона є джерелом різноманітних загроз для безпеки інформації.Перш за все, електронну пошту можна використовувати для розсилки спаму та фішингових повідомлень.

Спам - це небажані повідомлення, які надходять на поштову скриньку великою кількістю і зазвичай містять рекламу товарів або послуг.

Фішинг - це вид мошенництва, при якому зловмисники відправляють повідомлення, які намагаються викликати довіру, і запрошують користувачів на підроблені веб-сторінки для введення особистої інформації, такої як паролі та номери кредитних карток.

Фішингові атаки – не нове явище, але кіберхакери дуже активізувалися під час пандемії COVID-19, коли співробітники компаній вийшли за межі безпечного периметра організації і почали працювати віддалено. З початком повномасштабної війни IT-безпека бізнесу перевіряється на міцність ворожими кібервійськами, які будь-якими способами, у тому числі фішингом, намагаються завадити роботі компаній та отримати доступ до конфіденційної інформації.Користувач отримує лист, замаскований під лист від партнера, колеги, клієнта, у якому міститься посилання на фішинговий сайт. Нічого не підозрюючи, він клікає на нього. У цей момент відбувається крадіжка особистих даних, поширення шкідливого коду на пристрій з метою майбутньої крадіжки або шпигунства.

Те саме може статися, коли користувач отримує лист зі шкідливим вкладенням, наприклад, Excel-документом з макросом. Відкриваючи його, він запускає шкідливу програму, яка може порушити роботу як одного користувача, так і цілої компанії, поширившись у корпоративній мережі.

Загрози інформаційної безпеки, через користування ресурсами мережі інтернет показані на Рисунку 1.1.

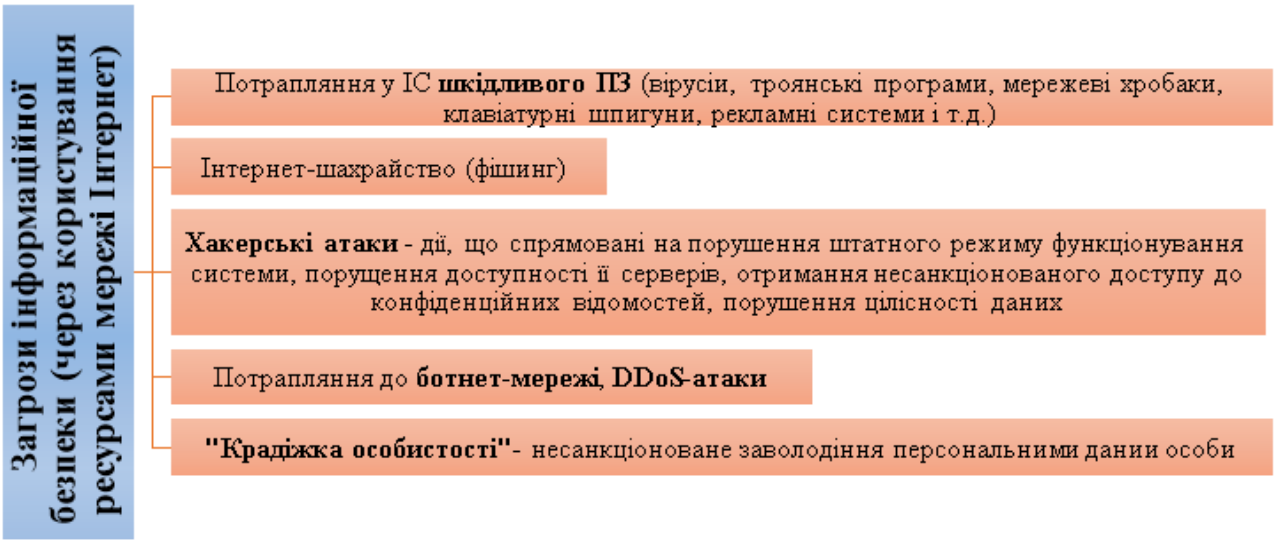


Рисунок 1.1. Загрози інформаційної безпеки, через користування ресурсами мережі інтернет

Шкідливе програмне забезпечення, основна мета якого – заволодіння потужностями комп’ютера та/або інформацією, яка там зберігається, з ціллю несанкціонованого використання комп’ютера та комп’ютерних систем. Типи шкідливих програм включають віруси, черв’яки, програми-вимагачі та програми-шпигуни.

Програма-вимагач — це кібератака, яка знищує або шифрує файли та папки, не дозволяючи власнику порушеного пристрою отримати доступ до своїх даних. Потім кіберзлочинець може вимагати гроші у власника бізнесу за ключ для розблокування зашифрованих даних. Але навіть у разі оплати кіберзлочинці можуть не надати ключа для повернення доступу.

Шахраї використовують спам для поширення шкідливих програм, обманом змушують одержувачів розголошувати конфіденційну інформацію чи

вимагають гроші.Наявність поштових застосунків практично на кожному комп'ютері і використання шкідливими програмами вмісту електронних адресних книг для виявлення нових жертв забезпечують сприятливі умови для розповсюдження шкідливих програм. Користувач зараженого комп'ютера, сам того не підозрюючи, розсилає заражені листи адресатам, які у свою чергу відправляють нові заражені листи і т.д. Нерідкісні випадки, коли заражений файл-документ унаслідок недогляду потрапляє в списки розсилки комерційної інформації якої-небудь крупної компанії. В цьому випадку страждають сотні або навіть тисячі абонентів таких розсилок, які потім розішлють заражені файли десяткам тисячам своїх абонентів.

У результаті цих загроз може бути порушена конфіденційність даних, пошкоджені файли або втрачена доступність до важливих даних. Безпека електронної пошти є дуже важливим аспектом забезпечення безпеки в Інтернеті.

1.1.2 Заходи щодо захисту скриньок від онлайн-загроз

Електронні листи - один з найбільш розповсюджених методів сучасного спілкування. І не дивно, адже, листування за допомогою електронної пошти дуже зручний спосіб передачі даних. Окрім того, сучасні поштові сервіси надають надійні методи превентивного захисту користувачів. Втім, варто пам’ятати, що не дивлячись на всі засоби захисту саме email листи залишаються, найбільш улюбленим інструментом кібершархаїв. Електронна пошта є важливою частиною сучасного життя більшості людей. Однак далеко не всі знають, що безпека електронних листів залежить не тільки від складності та унікальності пароля до облікового запису. Для захисту поштової скриньки від сучасних загроз необхідно також забезпечити:

- захист вмісту повідомлення під час передачі;
- перевірку вмісту листів на наявність посилань на шкідливі сайти та спам;

- аутентифікацію та авторизацію для безпечного доступу до облікових записів;
- цілісність та функціональність поштової програми.

Захист вмісту листів як основа безпеки пошти

При надсиланні повідомлення листи можуть легко опинитися під контролем зловмисників. Тому існує кілька додаткових рівнів захисту вмісту та процесу відправки повідомлень. Одним з найпоширеніших видів є шифрування листів на транспортному рівні, який забезпечує безпеку повідомлення в процесі передачі через Інтернет. Даний метод схожий на вкладення листа в конверт, тобто можна побачити звідки та куди направляється повідомлення, проте його зміст можна побачити тільки після відкриття так званого «конверта».

Інший метод захисту пошти — повне шифрування даних. Сутність даного методу полягає в тому, що повідомлення шифрується у відправника, а розшифровується в одержувача, при цьому в процесі передачі воно має зашифрований вигляд.

Захист пошти від спаму

Досить часто через електронну скриньку поширюється спам та інфіковані повідомлення, які містять шкідливе програмне забезпечення.

Спам – це надсилання великої кількості небажаних повідомлень. Найпоширенішим шляхом розповсюдження спаму є розсилання через електронну пошту: багато хто отримує щодня на e-mail листи з комерційними пропозиціями або взагалі з безглуздими текстами. Проте спамити можуть і через надокучливі смс, повідомлення в месенджерах, які не мають ніякої користі, через телефонні дзвінки тощо. Головна ознака спаму полягає в тому, що адресат, не надавав відправнику ані своїх даних (адреси пошти, телефонного номеру), ані згоди на отримання будь-яких повідомлень. Але навіть якщо ви залишили свої контакти у якомусь магазині, а звідти надходять по сто повідомлень на день, це також уже можна розцінити як спам.

Тому для захисту пошти від великої кількості небажаних та шкідливих електронних листів більшість поштових серверів мають «чорний» список відомих відправників спаму та фішингу. Також можна фільтрувати повідомлення за типом вкладення або дозволяти відправлення тільки від перевірених джерел. Багато організацій, щоб забезпечити захист пошти своїх користувачів, перевіряють повідомлення на наявність шкідливих програм і вірусів, перш ніж вони поширяться через мережу.

Щоб убезпечити себе від отримання спаму, необхідно наступне:

- Обачливо розповсюджуйте адресу своєї електронної пошти. Не публіковати її без потреби на публічних вебсайтах, форумах, сервісах. Особливо, якщо йдеться про e-mail, до якого прив'язані наші офіційні акаунти.
- За необхідності можна використовувати одноразові електронні адреси. Або зареєструвати кілька адрес під різні потреби. Скажімо, одну – для робочих контактів, іншу – для реєстрації на комерційних вебсторінках, форумах та інших ресурсах, де є підозра, що нашу пошту можуть використати для розсилання спаму.
- Не відповідати на спам, в жодному разі не відкривати посилань і вкладень, що надходять зі спамом.
- Сьогодні більшість поштових сервісів уже містять вбудований захист від спаму і відфільтровують небажані листи. Іноді до папки зі спамом може потрапити і важлива пошта. Тож періодично її можна перевіряти.
- Для захисту пристрою можна також використовувати антивірусне програмне забезпечення, яке має функцію розширеного захисту від спаму.

Безпека електронної пошти за допомогою авторизації та аутентифікації

Зловмисники часто використовують простий метод для поширення загроз через електронну пошту — маскування шкідливих листів під легітимні. Однак існують способи обмежити цю діяльність, хоч поки вони недостатньо поширені. Зокрема, ці методи допомагають перевіряти справжність вмісту

повідомлення та контролювати користувачів і облікові записи, яким дозволено відправляти повідомлення з домену нашої організації.

Використання аутентифікації та авторизації є важливою частиною управління поштовим сервером, як і швидке видалення акаунтів або, принаймні, зміна паролів облікових записів, які більше не використовуються. Наприклад, це відноситься до акаунтів, що раніше належали співробітникам, які більше не працюють в компанії.

Безпека облікового запису

Багатофакторна аутентифікація — це метод аутентифікації (ідентифікації), який вимагає від користувача надання двох або більше доказів особистості, щоб отримати доступ і увійти у свій обліковий запис. І тільки після введення всієї цієї необхідної інформації ми отримуємо доступ до свого облікового запису.

Багатофакторна аутентифікація — це один з найефективніших рівнів захисту пошти та доступу до облікових записів. Ідентифікація особистості здійснюється за допомогою одноразового ключа, який відправляється в SMS-повідомленні. Це може бути номер телефону, адреса електронної пошти або відповідь на якесь (відоме лише нам) секретне питання. Багатофакторна аутентифікація забезпечує захист пошти та є важливою частиною процесу входу в електронну скриньку або в мережу.

Захист програмного забезпечення

Також для посилення захисту електронної скриньки важливо регулярно оновлювати програмне забезпечення, зокрема, операційну систему, а також додаток або браузер, які ми використовуємо длядоступу, до електронної пошти. Це допоможе усунути уразливості, які дозволяють зловмисникам отримати доступ до наших електронних листів. Тому потрібно налаштувати автоматичне оновлення в програмному забезпеченні, операційній системі або безпосередньо на сайті постачальника.

1.1.3 Огляд методів захисту електронної пошти

В умовах широкого поширення глобальної мережі Інтернет в сучасному світі, зокрема на підприємствах та установах найвищого рівня, значення електронної пошти, як засобу швидкого пересилання повідомлень, документів, графічних, аудіо- та відеоматеріалів складно переоцінити. Надійність захисту даних в системі електронної пошти має безпосередній захист на загальний рівень інформаційної безпеки організації і, як наслідок, на ефективність її діяльності, що обумовлює важливість створення надійного захисту цього виду комунікацій.

Спам, лавинне розсилання, витік конфіденційної інформації - основні проблеми, з якими зустрічаються користувачі електронної пошти - пов’язані з недостатнім рівнем захисту сучасних поштових систем.

Розробники систем, спрямованих на захист електронної пошти, на власному досвіді знають, що миттєве вирішення проблеми захисту таких систем неможливе, оскільки хакери, творці та розповсюджувачі вірусів винахідливі, що спонукає постійно розвивати та вдосконалювати методи захисту. Слід також враховувати, що для забезпечення найвищого рівня захисту, необхідно застосовувати комплексний та систематичний підхід, з урахуванням всіх загроз та ризиків щодо безпеки пересилання електронних листів.

PGP (від англ. PrettyGoodPrivacy - «Досить хороша приватність») комп'ютерна програма, а також бібліотека функцій, що дозволяє виконувати операції шифрування та цифрового підпису повідомлень, файлів та іншої інформації, що представлена в електронному вигляді, в тому числі прозоре шифрування даних на запам’ятовуючих пристроях, наприклад, на жорсткому диску.

Перевагою такого методу є наявність сервера ключів keyserver.pgp.com., який дозволяє користувачам обмінюватись ключами та усуває необхідність публікації ключів, або передавати їх кожному адресату в особистому порядку.

До особливостей програми слід також віднести її спосіб захисту електронної пошти, точніше перехоплення трафіку поштового клієнта на рівні драйвера. Система опрацьовує трафік, шифрує повідомлення, що надсилаються і автоматично розшифровує вхідні повідомлення.

Цей метод має і свої недоліки, а саме: вже розшифровані повідомлення залишаються незахищеними в клієнті. Проблемою також є те, що якщо поштовий клієнт вже отримав повідомлення, а PGP Desktop не було запущено, то дешифрування листа стає непосильною задачею. Було розроблено спеціальні плагіни на такий випадок, але наразі їхня робота не є досить стабільною та задовільною.

S/MIME (від англ. Secure/Multipurpose Internet MailExtensions – «Безпечне/багатоцільове розширення для електронної пошти») — це стандарт для шифрування і підпису в електронній пошті за допомогою відкритогоключа. Під час роботи реалізується класична схема асиметричного шифрування з усіма її недоліками та перевагами, а саме: користувач генерує відкритий та закритий ключ, налаштовує свій поштовий клієнт і надсилає відкритий ключ всім бажаючим, які шифрують свої листи отриманим ключем і дешифруються лише закритим ключем.

Переваги S/MIME:

- листи в поштовому клієнті лишаються зашифрованими до тих пір, доки користувач сам їх не розшифрує. Для здійснення операції дешифрування необхідне введення паролю, що вказується під час створення ключової пари (відкритого/закритого ключа);
- на відміну від PGP Desktop, дешифрування відбувається поштовим клієнтом, а не окремою програмою, тому розшифрувати лист можна за будь-якої нагоди;
- підтримка більшості поштових клієнтів (в тому числі мобільних).

Недоліками є перш за все те, що постає питання про програму, яка згенерувала б сертифікат. Також необхідно обдумати питання реалізації обміну

ключами між учасниками. Потрібно також згадати про складнощі при зміні ключа, особливо, якщо користувачі не повністю розуміють суть своїх дій.

Однак, для нівелювання недоліків S/MIME було створено спеціальний плагін CyberSafe Mail, що дозволяє публікувати свій ключ на сервері ключів. Також є можливість пошуку ключів, опублікованих іншими користувачами. Втім, даний плагін доступний поки що лише для Microsoft Outlook.

HushMail – сервіс електронної пошти з шифруванням. Користується великим популярністю у недосвідчених користувачів, як захищений сервіс, що не потребує попередніх налаштувань і одразу готовий до роботи. Перевагами такої і подібних їй систем є простота у використанні та відсутність необхідності налаштувань. Недоліком же є здійснення криптографічних операцій на сервері.

Також для здійснення захисту електронної пошти застосовуються різні плагіни для браузерів:

1) плагін браузера PGP Mail. Даний плагін використовує асиметричне шифрування на стороні клієнта та підтримує браузери Firefox, Chrome, Opera, Safari. Недоліками є рекомендації щодо використання TOR (система забезпечення анонімності в мережі Інтернет), що не є зручним для недосвідчених користувачів та маленька кількість підтримуваних браузерів.

2) плагін браузера SecureGmail ввикористовує симетричне шифрування, що є зручним лише при спілкування з невеликою кількістю адресатів. Також цей плагін працює лише з браузером Chrome.

3) плагін браузера Encrypted Communication за своїми можливостями схожий на SecureGmail, але працює лише в браузері Firefox.

Найпростіший спосіб захисту електронної пошти – використання симетричного шифрування. Для його реалізації можна використовувати плагіни браузера SecureGmail і Encrypted Communication. Для забезпечення більш надійного захисту конфіденційності даних в ідеалі рекомендується використовувати S/MIME, надійність якого полягає в тому, що повідомлення

зберігаються в поштовому клієнті в зашифрованому вигляді і розшифровуються лише при зверненні до них.

1.1.4 Протоколи роботи з електронною поштою

В Інтернеті для роботи з електронною поштою використовуються прикладні протоколи SMTP, POP, IMAP4.

Протокол SMTP (SimpleMassageTransferProtocol) — простий протокол, що підтримує передачу повідомлень між будь-якими вузлами Інтернет. Маючи механізми проміжного збереження пошти і підвищення надійності доставки, протокол SMTP припускає використання різноманітних транспортних служб і поштових серверів. Він може працювати навіть в мережах, які не підтримують стек протоколів TCP/IP. Протокол SMTP дозволяє групувати повідомлення на адресу одного одержувача і розмножувати копії E-mailповідомлення для передачі за різними адресами.

Протокол POP (PostOfficeProtocol) надає кінцевому користувачу можливість доступу до його електронних повідомлень. POP-клієнти при запиті користувача на отримання пошти вимагають ввести пароль, що підвищує конфеденційність листування.

Для невеликих організацій невигідно тримати у себе систему для передачі повідомлень (messagetransportsystem). Це пов'язано з тим, що в невеликих організаціях, які не спеціалізуються на комп'ютерних технологіях, як правило, робочі станції клієнтів мережі не мають достатньо ресурсів (продуктивність або дисковий простір) для забезпечення роботи повного SMTP-серверу. Крім того, таким користувачам електронної пошти може бути невигідним тримати персональний комп'ютер, постійно підключений до Інтернету. Для вирішення цієї проблеми і був створений поштовий протокол для роботи в офісі — POP (PostOfficeProtocol). Його найпоширеніший варіант — POPvS (протокол поштового відділення версії. Цей протокол дозволяє робочим

станціям динамічно мати доступ до свого поштового ящика, розташованого на сервері, який призначений для обслуговування електронної пошти в даній організації. POPvS — це найпростіший протокол для роботи користувача зі змістом своєї поштової скриньки. Він дозволяє тільки забрати пошту з поштової скриньки серверу на робочу станцію клієнта і знищити її з поштової скриньки на сервері.

Всю подальшу обробку поштове повідомлення проходить на комп'ютері клієнта. POP-сервер не відповідає за відправку пошти, він працює тільки як універсальна поштова скринька для групи користувачів. Якщо користувачу необхідно відправити повідомлення, він повинен встановити з'єднання з SMTP-сервером і відправити туди свої повідомлення за SMTP-протоколом.

Протокол IMAP4 (Internet Message Access Protocol, Version дозволяє клієнтам діставати доступ і маніпулювати повідомленнями електронної пошти на сервері. Протокол IMAP4 відрізняється від протоколу POPvS тим, що IMAP4 підтримує роботу з системою каталогів (або папок) повідомлень.

IMAP4 дозволяє керувати каталогами (папками) віддалених повідомлень так само, якби вони розташовувалися в локальному комп'ютері. IMAP4 дозволяє клієнту створювати, видаляти і перейменовувати поштові скриньки, перевіряти наявність нових повідомлень і видаляти старі. Завдяки тому, що IMAP4 підтримує механізм унікальної ідентифікації кожного повідомлення в поштовій папці клієнта, він дозволяє читати з поштової скриньки тільки повідомлення, які задовольняють певним умовам, або їх частині, змінювати атрибути повідомлень і переміщати окремі повідомлення.Взаємодія протоколів роботи електронної пошти показана на Рисунку 1.2.

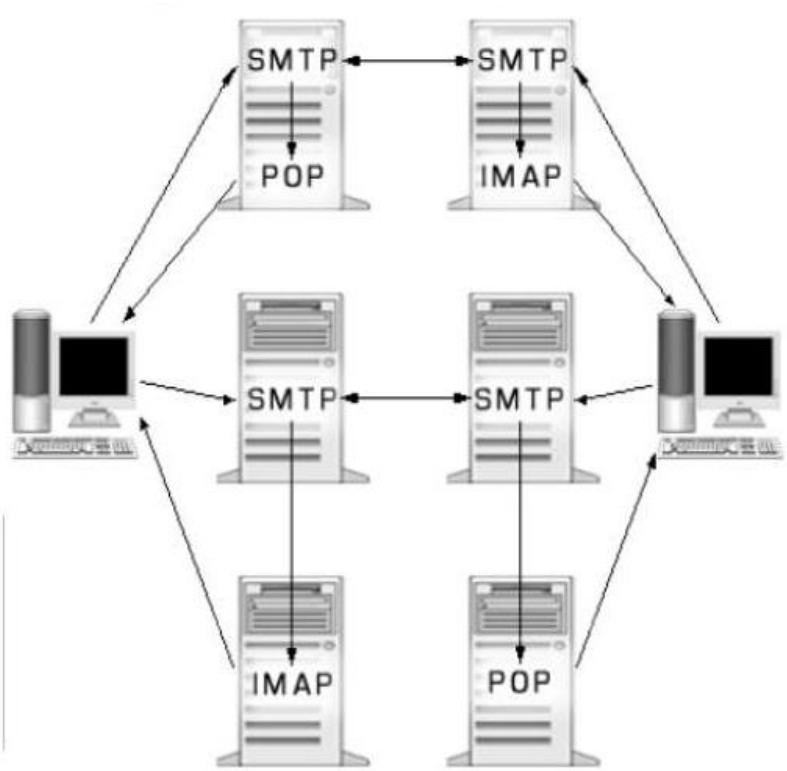


Рисунок 1.2 Взаємодія протоколів роботи електронної пошти

1.2 Алгоритми шифрування

Шифрування — оборотне перетворення інформації з метою приховування від неавторизованих осіб з наданням авторизованим користувачам доступу до неї. Головним чином шифрування служить завданням дотримання конфіденційності інформації, що передається. Важливою особливістю алгоритму шифрування є використання ключа, який затверджує вибір конкретного перетворення із сукупності можливих даного алгоритму.

SSL (SecureSocketsLayer - рівень захищених сокетів) - криптографічний протокол, який передбачає більш безпечний зв'язок. Він використовує асиметричну криптографію для автентифікації ключів обміну, симетричне шифрування збереження конфіденційності, коди автентифікації повідомлень для цілісності повідомлень. Протокол широко використовувався для обміну миттєвими повідомленнями у таких додатках, як електронна пошта, інтернет-

факс та ін. Головна мета SSL сертифікатів – шифрування інформації під час передачі її по інтернету від точки А в точку Б. Тобто, всі дані, будь-то особиста інформація, банківські координації, геолокація, зображення та інші важливі відомості, які можуть використовувати зловмисники, буде шифруватися і доставлятися до одержувачу в безпеці, без втручання з боку.

1.2.1 Симетричне шифрування

Метод симетричного шифрування, (рис.1.3), як і випливає з назви, використовує один криптографічний ключ для шифрування і дешифрування даних. Використання одного ключа для обох операцій робить процес простим.

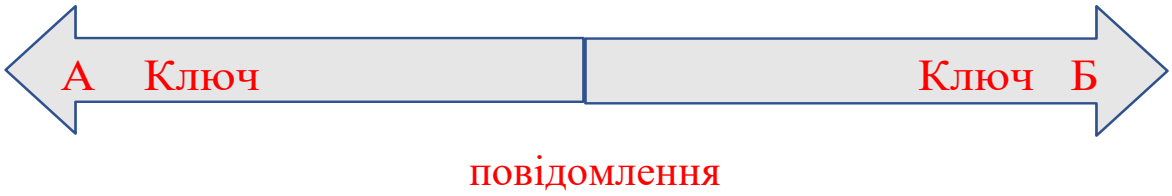


Рисунок 1.3 Приклад симетричного шифрування

Наприклад щоб захистити лист, повідомлення шифрується таким чином, щоб кожна буква замінювалася буквою на чотири позицій вниз за алфавітом. Замість того, щоб писати «Car», пишемо «Gev» (C ->G, A ->E, R ->V). Щоб розшифрувати повідомлення, необхідно кожну букву замінити на чотири позицій в алфавітному порядку назад. Таку техніку шифрування використовував римський імператор і військовий генерал Гай Юлій Цезар, відома вона ще як «шифр Цезаря».

Найбільш видатною особливістю симетричного шифрування є простота процесу, так як використовується один ключ як для шифрування, так і для дешифрування. Там, де необхідно зашифрувати великий шматок даних, симетричне шифрування виявляється відмінним варіантом. В результаті алгоритми симетричного шифрування:

- Значно швидше, ніж їх аналоги асиметричного шифрування (про що ми незабаром поговоримо);

- Потребують менше обчислювальної потужності;
- Не знижується швидкість інтернету.

“Шифр Цезаря” засновано на особливій логіці шифрування даних, розгадавши яку можна легко розшифрувати інформацію. Сучасні ж методи шифрування, засновані на дуже складних математичних функціях, які зламати практично неможливо. Існують сотні алгоритмів симетричного типу, найбільш поширені з них – AES, RC4, DES, 3DES, RC5, RC6 і т. д. Розглянемо три найбільш популярних алгоритми.

1.2.2 Асиметричне шифрування

Асиметричне шифрування, на відміну від симетричного, включає в себе кілька ключів для шифрування і дешифрування даних, які математично пов’язані один з одним. Один з цих ключів відомий як «відкритий ключ», а інший – як «закритий ключ». Асиметричний метод шифрування також відомий як «криптографія з відкритим ключем».(рис. 1.4)

Рисунок 1.4 Асиметричний метод шифрування

При шифруванні з відкритим ключем, відкритий ключ надаєтьсякожному, хто відправляє інформацію, а секретний ключ зберігається при собі. Шифрування інформації проходить за допомогою відкритого ключа, щоб дані можна було розшифрувати тільки за допомогою особистого ключа. Це виключає ризик компрометації ключа, оскільки дані можуть бути розшифровані тільки з використанням закритого ключа.

Перша (і найбільш очевидна) перевага цього типу шифрування – безпека, яку він забезпечує. У цьому методі відкритий ключ – який є загальнодоступним

– використовується для шифрування даних, в той час як розшифрування даних виконується з використанням закритого ключа, який необхідно надійно зберігати. Це гарантує, що дані залишаються захищеними від атак «людина посередині» (MiTM). Для веб-серверів і серверів електронної пошти, які постійно підключаються до сотень тисяч клієнтів потрібно управляти тільки одним ключем і захищати його. Інший ключовий момент полягає в тому, що криптографія з відкритим ключем дозволяє створювати зашифроване з'єднання без необхідності зустрічатися в автономному режимі, щоб спочатку обмінятися ключами.

Друга важлива особливість, яку пропонує асиметричне шифрування, – це аутентифікація. Як ми бачили, дані, зашифровані за допомогою відкритого ключа, можуть бути розшифровані тільки за допомогою закритого ключа, пов'язаного з ним. Отже, він гарантує, що дані бачить і дешифрує тільки той об'єкт, який повинен їх отримати. Простіше кажучи, це підтверджує, що ви розмовляєте або обмінюєтеся інформацією з реальною людиною або організацією. Розглянемо два основних типи алгоритмів асиметричного шифрування.

1.2.3 Гібридне шифрування

Гібридна схема шифрування - це поєднання зручності асиметричної схеми шифрування з ефективністю симетричної схеми шифрування. Гібридне шифрування не є «окремим методом», як симетричне або асиметричне, в ньому використовуються всі переваги обох методів і створюється синергія надійних систем шифрування.

У гібридному шифруванні симетричний ключ використовується для шифрування даних, а потім цей ключ шифрується асиметричним ключем та відправляється разом з зашифрованими даними. Одержувач використовує свій

приватний ключ для розшифрування симетричного ключа, а потім використовує цей симетричний ключ для розшифрування даних.

Кожен з алгоритмів шифрування має свої недоліки. Наприклад, метод симетричного шифрування відмінно підходить для швидкого шифрування великих обсягів даних. Але він не забезпечує перевірку особистості, що є необхідним, коли мова заходить про безпеку в Інтернеті. З іншого боку, асиметричне шифрування надає доступ до даних передбачуваного одержувача. Однак ця перевірка робить процес шифрування занадто повільним.

Ідея гібридного шифрування народилася, коли стало критично важливо шифрувати дані з високою швидкістю надаючи при цьому перевірку особистості.Метод гібридного шифрування використовується в SSL / TLS сертифікатах під час послідовного зв’язку між серверами і клієнтами (веб-браузерами) в процесі, відомому як “TLS handshake”. Спочатку перевіряється особистість обох сторін з використанням закритого і відкритого ключа. Після того, як обидві сторони підтвердили свою особистість, шифрування даних відбувається за допомогою симетричного шифрування з використанням ефемерного (сеансового) ключа. Це забезпечує швидку передачу великого обсягу даних, які ми відправляємо і отримуємо в Інтернеті кожну хвилину.

Поєднання методів шифрування має різні переваги. Одне полягає в тому, що канал зв'язку встановлюється між двома наборами обладнання користувачів. Потім користувачі мають можливість спілкуватися за допомогою гібридного шифрування. Асиметричне шифрування може уповільнити процес шифрування, але при одночасному використанні симетричного шифрування посилюються обидві форми шифрування. Результатом є додаткова безпека передавального процесу разом із загальною покращеною продуктивністю системи.

Такий підхід є безпечнішим, ніж використання лише симетричного шифрування, оскільки він дозволяє використовувати більш складні та надійні асиметричні шифри для передачі симетричного ключа, що може бути небезпечним у відкритій мережі. Крім того, гібридне шифрування дозволяє

використовувати ефективні та швидкі симетричні шифри для шифрування даних, що дозволяє забезпечувати швидку передачу даних.Порівняльна характеристика симетричного та асиметричного методів шифрування показано в Таб.1.

Таблиця 1. Порівняльна характеристика симетричного та асиметричного методів шифрування

Симетричне шифрування	**Асиметричне шифрування**
Один ключ використовується для шифрування і дешифрування даних.	Пара ключів використовується для шифрування і дешифрування. Ці ключі відомі як “відкритий ключ” і “закритий ключ”.
Простий метод шифрування, так як використовується тільки один ключ.	У зв’язку з тим, що використовується пара ключів – процес складний.
Використовується для шифрування великих об’ємів даних.	Забезпечує аутентифікацію.
Забезпечує високу продуктивність і вимагає менше обчислювальної потужності.	Складні процеси протікають повільніше і вимагають більшої обчислювальної потужності.
Для шифрування даних використовується менша довжина ключа (128-256 біт).	Використовуються довші ключі шифрування (1024-4096 біт).
Ідеально підходить для шифрування великої кількості даних.	Використовується при шифруванні невеликого об’єму даних.
Стандартні алгоритми: RC4, AES, DES, 3DES і QUAD.	Стандартні алгоритми: RSA, Diffie-Hellman, ECC, ElGamal і DSA.

Більшість сучасних SSL сертифікатів використовують гібридний метод: асиметричне шифрування для аутентифікації і симетричне шифрування для конфіденційності. Такий сертифікат не дає шахраям перехопити або підмінити

особисті дані користувачів: контактну інформацію, номери банківських карт, логіни, паролі, адреси електронної пошти і т.д.

1.3 Розробка та реалізація алгоритму шифрування

1.3.1 Шифрування електронної пошти

Коли ми надсилаємо повідомлення електронної пошти, воно проходить кілька етапів, щоб переконатися, що його зможе прочитати лише адресат. Перший етап називається "шифрування", він зашифровує повідомлення так, що розшифрувати його може тільки той, хто має доступ до потрібного ключа. Другий етап називається "розшифрування", і він дозволяє одержувачам отримати свої оригінальні повідомлення без необхідності проходити спочатку етапи шифрування - їм просто потрібно знати, де зберігаються ключі, щоб вони могли отримати власні повідомлення.

Шифрування – це процес кодування повідомлення таким чином, щоб його могли прочитати лише авторизовані користувачі. Мета шифрування електронної пошти – переконатися, що ваші листи захищені від сторонніх очей і можуть бути прочитані лише тими, кому ви довіряєте.Шифрування електронної пошти: Процес кодування інформації у формат, який не може бути прочитаний ніким, крім тих, хто має доступ до ключа чи пароля (зазвичай це імена користувачів та паролі). Послуги шифрування електронної пошти також називають послугами наскрізного шифрування, оскільки вони шифрують усі повідомлення, що надсилаються між відправником та одержувачем, так що розшифрувати їх можуть лише ті, хто має доступ до відповідних ключів. Розглянемо типи шифрування електронної пошти.

1. Шифрування електронної пошти S/MIME

S/MIME розшифровується як Secure/Multipurpose Internet MailExtensions. Це спосіб зашифрувати повідомлення так, щоб його міг отримати лише гаданий

одержувач. Він корисний для конфіденційних даних, таких як номери кредитних карток, номери соціального страхування та інформація про банківські рахунки. Цей процес складається із двох частин: в одній частині повідомлення шифрується, а в іншій – розшифровується. Шифрування відбувається перед тим, як одержувач отримує повідомлення; потім він розшифровує його перед прочитанням.

2. Шифрування електронної пошти PGP

PGP (PrettyGoodPrivacy) – це вид шифрування з відкритим ключем, що використовується тими, яким є що приховувати від недоброзичливців, які можуть захотіти перехопити їхні повідомлення чи інші цифрові комунікації.PGP працює шляхом створення цифрового підпису на кожному повідомленні, яке вимагає відкритих ключів обох сторін перед обміном повідомленнями через Інтернет. При використанні цього методу шифрування будь-яке повідомлення може бути прочитане лише тим, хто має доступ до вашого відкритого ключа та знає секретну ключову фразу, яка розблокує цю пару ключів. Це робить його надзвичайно безпечним – навіть якщо хтось зламає ваш закритий ключ, він не зможе прочитати жодного з ваших повідомлень.

3. Безпека транспортного рівня (TLS)

TransportLayer Security, або TLS, - це протокол, який використовується для шифрування зв'язку між клієнтом та сервером.TLS забезпечує безпеку з'єднання між веб-браузером та веб-сайтом. TLS дозволяє захистити конфіденційні дані, які передаються між браузером та веб-сайтом, від перехоплення та зловмисного використання. TLS використовується для захисту різних типів даних, таких як особисті дані, банківські дані, конфіденційна інформація про бізнес тощо. Він також використовується для підключення до веб-сайту за допомогою HTTPS. Цешифрування означає, щоколи ми надсилаємо повідомлення електронної пошти, безпека транспортного рівня захищає наше повідомлення від прочитання будь-ким, хто перехоплює його під час

проходження через Інтернет. Це допомагає захистити повідомлення від перехоплення хакерами або службами спостереження.

TLS працює на рівні транспортного протоколу, тому він може захищати дані, що передаються через будь-який протокол, що працює на транспортному рівні, наприклад, HTTP, SMTP, FTP, POP3 та інші.TLS забезпечує безпеку з'єднання за допомогою криптографічних протоколів. Для захисту даних TLS використовує симетричне шифрування та асиметричне шифрування, що забезпечує конфіденційність, цілісність та аутентичність даних.

Тема безпеки електронних пристроїв стала все більш актуальною в останні роки, оскільки зростає кількість крадіжок інформації та персональних даних з пристроїв, таких як телефони та комп'ютери. Якщо хтось украде наш пристрій, то це може мати серйозні наслідки для нашої конфіденційної інформації, включаючи електронні листи та інші особисті дані.

Хоча безпека на транспортному рівні може захистити наші електронні листи, якщо ми використовуємо захищені протоколи та веб-сайти, що використовують шифрування, відсутність фізичного доступу до наших пристроїв є ключовим фактором безпеки. Якщо зловмисник отримає фізичний доступ до нашого пристрою, то він може відновити паролі, отримати доступ до нашої електронної пошти та іншої конфіденційної інформації.

1.3.2 Алгоритм шифрування Цезаря

Шифр Цезаря – один з найбільш простих та широковідомих алгоритмів шифрування текстових даних. Цей метод названий в честь римського полководця Гая Юлія Цезаря, який використовував шифр для приватного листування з підлеглими. Шифр Цезаря - це тип шифру підстановки, де кожна буква звичайного тексту замінюється буквою з фіксованим числом позицій вниз по алфавіту. Принцип його дії можна побачити на Рисунку1.5.

Алгоритм шифрування Цезаря - це метод шифрування повідомлення шляхом заміни кожної букви відкритого тексту на букву, що знаходиться за k позицій в алфавіті (k - ключ шифрування). Наприклад, при k=3 буква A замінюється на D, B на E і т.д

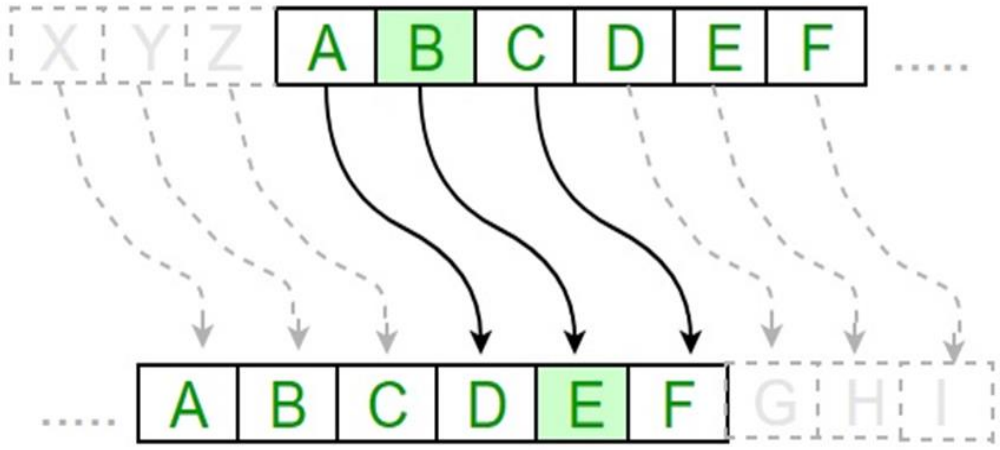


Рисунок 1.5 Принцип дії Шифра Цезаря

Алгоритм шифрування Цезаря полягає заміні кожного символу вхідного повідомлення на символ, який знаходиться на деякій константній відстані з правої чи лівої сторони. Відстань при цьому називають – ключем. На рисунку 1.5 показаний приклад для ключа 3, та отримаємо послідовність:

Алфавіт: A B C D EFG H I J K L M N O P Q R S T U V W X Y Z

Шифр: 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

Тобто A замінюємо на D, B на E, і т. д.

шифр Цезаря можна описати наступними формулами:

y = (x + k);

x = (y – k).

де x - відкритий текст, k - ключ шифрування, y - зашифрований текст.

Історично Шифр Цезаря можна назвати - шифром зрушення, з тієї причини, що алгоритм шифрування може інтерпретуватися як "клавіша зсуву букви вниз", а алгоритм дешифрування може інтерпретуватися як "клавіші зсуву букви вгору". Наприклад, якщо ключ= 15, Алгоритм кодування зрушує букву на15букв вниз (до кінця алфавіту). Алгоритм дешифрування зрушує букву

на15букв вгору (до початку алфавіту). Звичайно, коли ми досягаємо кінця або початку алфавіту, ми рухаємося по кільцю до початка (оголошені властивості операції по модулю 26).Розглянемо приклад шифру Цезаря для українського тексту:

Алфавіт:

А Б В Г Ґ Д Е Є Ж З И І Ї Й К Л М Н О П Р С Т У Ф Х Ц Ч Ш Щ Ь Ю Я

Шифр:

Д Е Є Ж З И І Ї Й К Л М Н О П Р С Т У Ф Х Ц Ч Ш Щ Ь Ю Я А Б В Г Ґ

Припустимо, необхідно зашифрувати фразу " **нам потрібна допомога** "

Для ключа 5 отримаємо послідовність " **тдс фучхметд иуфусужд** "

Як ми бачимо, фраза перетворюється на набір букв, які дуже легко розшифрувати знаючи, що ключ 5.

Такі шифри вразливі до атак тільки зашифрованого тексту, коли використовується вичерпний перебір ключів (атака грубої сили). Безліч ключів адитивного шифру дуже мало - їх тільки33. Один з ключів, нульовий, є марним (зашифрований текст буде просто відповідати початкового тексту). Отже, залишається тільки32можливих ключів. Отже може легко почати атаку грубої сили зашифрованого тексту і досягти успіху.

Отримавши доступ до зашифрованого тексту " тдсфучхметдиуфусужд " можливо зламати шифр, використовуючи атаку грубої сили.

Рішення: Необхідно послідовно перебирати ключі починаючи з першого.За допомогою ключа номер 5 отримуємо осмислений текст "нам потрібна допомога"

Шифрування Цезаря є дуже простим методом шифрування, але він також є дуже слабким, оскільки його можна легко зламати шляхом перебору всіх можливих ключів. Тому, використовувати його для захисту важливих даних не рекомендується.

1.3.3 Алгоритм шифрування Скитала

Шифр Скитала – це шифрування тексту за допомогою дерев'яного циліндру та пергаменту, також відомий як шифр Давньої Спарти. Цей метод шифрування використовувався античними спартанцями, під час війни.

Для шифрування тексту використовується циліндр фіксованого діаметру, на який намотується вузька полоска пергаменту. Пергаментна стрічка намотувалась на циліндр так, щоб не було ні просвітів, ні нахльостів. Повідомлення записують вздовж циліндру, а потім розмотують, після чого отримуємо зашифроване повідомлення, яке можна розшифрувати використавши циліндр такого ж діаметру(рис.1.6). При цьому діаметр циліндра виступає в якості ключа шифрування.

Рисунок 1.6 Циліндршифрування тексту алгоритму Скитала

Наприклад, використовується паличка, по довжині кола якої міститься 4 символи (число рядків у таблиці), а довжина самої палички дозволяє записати 5 символів (число стовпців у таблиці), вихідний текст «**шифр давньоїспарти**». Схематично це можна зобразити як показано в Таб 2:

Таблиця 2. Таблиця шифрування алгоритму Скитала

Ш	И	Ф	Р	-
Д	А	В	Н	Ь
О	Ї	-	С	П
А	Р	Т	И	-

Після розмотування стрічки, шифротекст буде наступним «ШДОАИАЇРФВ_ТРНСИ_ЬП»

Фактично такий шифр передбачає, запис тексту по рядках із заздалегідь відомою кількістю рядків а зчитування (шифрування) по стовпцях. Зворотна процедура спричинить розшифрування тексту.

Параметри шифру Скитала позначаються як: n – кількість стовпців і m – кількість рядків. Оскільки повідомлення часто мають різну довжину, обидва ці параметри за незмінний ключ взяти незручно. Тому зазвичай як відомий кожній стороні ключа вибирається один з них, часто це m, а другий обчислюється на основі відомого і довжини повідомлення: n=[(L-1)/m]+1}, де [(L-1)/ m] – ціла частина числа, а L–довжина повідомлення.

1.3.4 Розробка алгоритму шифрування

Зважаючи на переваги та недоліки розглянутих вище алгоритмів шифрування, робимо висновок, що кожен з них можна застосувати для шифрування e-mail повідомлень. Насамперед через їхню простоту реалізації. Звичайний користувач, що розібрався з роботою даних шифрів, може самостійно зашифрувати ними своє вихідне повідомлення і проінструктуючи одержувача, створити умови для розшифровки. Але як показав криптоаналіз у попередніх розділах, за рахунок відносно небагатьох можливих ключів, їх застосування поодинці неефективне. У разі отримання доступу до повідомлення, будь-який з таких шифрів можна швидко розкрити.

Блок-схема алгоритму шифрування

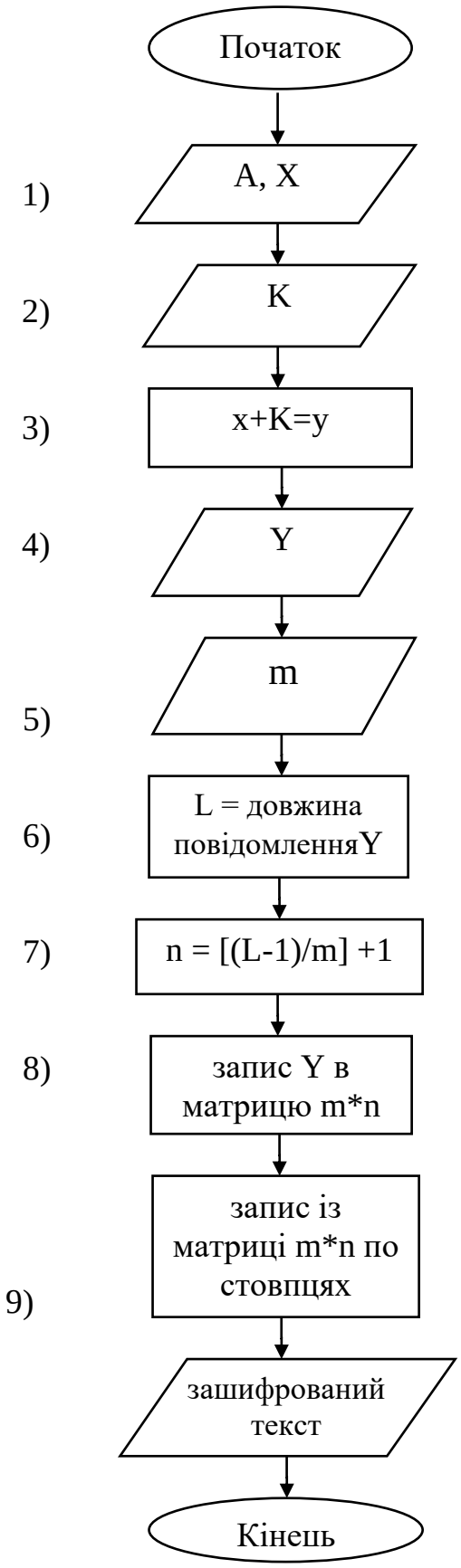


Рисунок 1.7 Блок-схема алгоритму шифрування

Однак складність розшифрування можна підвищити за рахунок спільного використання алгоритму шифрування Цезаря та алгоритма шифрування Скитала (рис.1.7). Такий варіант шифрування також можна легко здійснити не застосовуючи технічних засобів, зате процес дешефрування може зайняти набагато більше часу і навіть не увінчатися успіхом, якщо не знати принципу шифрування.

Шифрування проводиться таким чином:

1) A - визначаємо алфавіт, вводимо відкритий текст X.

2) Вводимо ключ шифрування K (усунення при шифруванні Цезаря).

3) Для отримання символів yi зашифрованого повідомлення (шифром Цезаря), зсуваємо символихі вихідного повідомлення на k позицій праворуч.

4) Визначаємо зашифроване повідомленняY (шифром Цезаря), яке далі шифруватиметься шифром Скитала.

5) Вводиться "ключ" m - кількість рядків матриці Скитала.

6) Обчислюється L – довжина повідомлення Y.

7) Обчислюється кількість стовпців n за формулою [(L-1)/m]+1

8) Записуємо повідомлення Y в матрицю m*n.

9) Вміст матриці m*n зчитується послідовно стовпцями зверху вниз. Отримуємо результуючий зашифрований текст.

1.3.5 Шифрування розробленим алгоритмом

1) На початку реалізації мого шифру визначаємо алфавіт вихідного повідомлення. У нашому випадку алфавіт Український:

А Б В Г Ґ Д Е Є Ж З И І Ї Й К Л М Н О П Р С Т У Ф Х Ц Ч Ш Щ Ь Ю Я

Наступний крок – вводимо текст повідомлення, яке плануємо зашифрувати, де X1, X2, X3, X4, X5

X1 вітаюшановний пан адвокат

X2 надсилаю вам пароль від мого банківськогоосередку

X3 сім вісім три один три вісім

X4 та відповідь на секретне запитання

X5 це слово холодець

2) На кроці вибору ключа шифру Цезаря, є два варіанти реалізації цієї дії:

- ключ буде за початковою домовленістю постійним
- визначається процедура передачі інформації про ключ на кожне повідомлення

Обидва варіанти не є оптимальними. У першому випадку ключ рано чи пізно може бути розкритий і зловмисник отримає постійний доступ до нашого листування.При другому варіанті виникає додаткове завдання щодо організації каналу зв'язку для передачі поточного ключа шифрування.

Беручи до уваги ці обставини, пропонується передавати ключ шифрування у рядку повідомлення.Така реалізація дає можливість не тільки змінювати ключ розшифровки кожного повідомлення, але й усередині листа шифрувати кожен рядок своїм власним ключем.Цей підхід у рази підвищує криптографічну стійкість алгоритму, а його криптоаналізвимагатиме значних витрат часу на розкриття.

Відповідно до нашого повідомлення, у першому рядку чотири слова, значить ключ шифруванняпершого рядка буде К=4

1 2 3 4

X1 вітаю шановний пан адвокат

У другому рядку сім слів, ключ шифрування другого рядка буде К=7

1 2 3 4 5 6 7

X2 надсилаю вам пароль від мого банківського осередку

У третьому рядку шість слів, та ключ шифрування цього рядка буде К=6.

1 2 3 4 5 6

X3 сім вісім три один три вісім

У четвертому рядку п'ять слів, отже ключ шифрування четвертого рядка буде К=5

1 2 3 4 5

X4 та відповідь на секретне запитання

У п'ятому рядку три слова, отже ключ шифрування п'ятого рядка буде *К=3*

1 2 3

X5 це слово холодець

3) Зсуваємо праворуч символи Xi кожного рядка на відповідний ключіз попереднього пункту.

x+ Ключ 4= y

А Б В Г Ґ Д Е Є Ж З И І Ї Й К Л М Н О П Р С Т У Ф Х Ц Ч Ш Щ Ь Ю Я

X_1 ВІТАЮ_ШАНОВНИЙ_ПАН_АДВОКАТ

4) Визначаємо зашифроване повідомлення Y

Y_1 ЕЛЦҐВ_ЯҐСТЕСКН_УҐС_ҐЗЕТОҐЦ

5) Кількість рядків матриці m також зашифровуємо в повідомленні. Зручно буде їх прирівняти до кількості рядків у листі і воно буде однаково для всіх рядків (зате змінюються в інших листах, при іншій кількості рядків у листі). Відповідно до нашого повідомлення m=5.

6) L_1 – довжина повідомлення Y_1 дорівнює кількості літер у кожному рядку з урахуванням прогалин між словами. У першому рядку нашого повідомлення L_1=26.

7) Обчислюється кількість стовпців n за формулою$[(L_1-1)/m] +1$,

де $[(L_1-1)/m]$ ціла частина числа:

кількість стовпців $n = [(L_1-1)/m]+1= (26-1)/5+1=6$

8) Записуємо повідомлення Y_1 в матрицю m*n, дем=5 та n=6. (Таб.3)

Y_1 ЕЛЦҐВ_ЯҐСТЕСКН_УҐС_ҐЗЕТОҐЦ

Табоиця 3. Матриця m*n (Y_1)

Е	Л	Ц	Ґ	В	_
Я	Ґ	С	Т	Е	С
К	Н	_	У	Ґ	С
_	Ґ	З	Е	Т	О
Ґ	Ц	_	_	_	_

9) Вміст матриці зчитується послідовно стовпцями зверху вниз. Отримуємо результуючий зашифрований текст (Y_1):

ЕЯК_ҐЛҐНҐЦЦС_З_ҐТУЕ_ВЕҐТ__ССО

Аналогічним чином шифруємо інші рядки і передаємо все повідомлення одержувачу.

Шифрування X_2

X_2 НАДСИЛАЮ_ВАМ_ПАРОЛЬ_ВІД_МОГО_БАНКІВСЬКОГО_ОСЕРЕДКУ Ключ 7

Y_2 ФЄЇШНТЄД_ЗЄУ_ЦЄЧХТҐ_ЗОЇ_УХИХ_ЖЄФСОЗШҐСХИХ_ХШЙЧЙЇСЬ

Ф	Є	Ї	Ш	Н	Т	Є	Д	_	З
Є	У	_	Ц	Є	Ч	Х	Т	Ґ	_
З	О	Ї	_	У	Х	И	Х	_	Ж
Є	Ф	С	О	З	Ш	Ґ	С	Х	И
Х	_	Х	Ш	Й	Ч	Й	Ї	С	Ь

Таблиця 4. Матриця m*n (Y_2)

Отримуємо результуючий зашифрований текст(Y_2):

ФЄЗЄХЄУОФ_Ї_ЇСХШЦ_ОШНЄУЗЙТЧХШЧЄХИҐЙДТХСЇ_Ґ_ХСЗ_ЖИЬ

Шифрування X_3

X_3 СІМ_ВІСІМ_ТРИ_ОДИН_ТРИ_ВІСІМ Ключ 6

Y_3 ЧНТ_ЖНЧНТ_ШЦМ_ФІМУ_ШЦМ_ЖНЧНТ

Таблиця 5. Матриця m*n (Y_3)

Ч	Н	Т	_	Ж	Н
Ч	Н	Т	_	Ш	Ц
М	_	Ф	І	М	У
_	Ш	Ц	М	_	Ж
Н	Ч	Н	Т	_	_

Отримуємо результуючий зашифрований текст(Y_3):

ЧЧМ_ННН_ШЧТТФЦН__ІМТЖШМ__НЦУЖ

Шифрування X_4

X_4 ТА_ВІДПОВІДЬ_НА_СЕКРЕТНЕ_ЗАПИТАННЯКлюч 5

Y_4 ЧД_ЄМИФУЄМИВ_ТД_ЦІПХІЧТІ_КДФЛЧДТТҐ

Таблиця 6. Матриця m*n (Y_4)

Ч	Д	_	Є	М	И	Ф
У	Є	М	И	В	_	Т
Д	_	Ц	І	П	Х	І
Ч	Т	І	_	К	Д	Ф
Л	Ч	Д	Т	Т	Ґ	_

Отримуємо результуючий зашифрований текст(Y_4):

ЧУДЧЛДЄ_ТЧ_МЦІДЄИІ_ТМВПКТИ_ХДҐФТІФ

Шифрування X_5

X_5 ЦЕ_СЛОВО_ХОЛОДЕЦЬКлюч 3

Y_5 ЩЗ_ФОСДС_ШСОСЖЗЩА

Таблиця 7. Матриця m*n (Y_5)

Щ	З	_	Ф
О	С	Д	С
_	Ш	С	О
С	Ж	З	Щ
А	_	_	_

Отримуємо результуючий зашифрований текст(Y_5):

ЩО_САЗСШЖ__ДСЗ_ФСОЩ

Запишемо повністювихіднета закодоване повідомлення:

Вихідне повідомлення:

ВІТАЮ_ШАНОВНИЙ_ПАН_АДВОКАТ

НАДСИЛАЮ_ВАМ_ПАРОЛЬ_ВІД_МОГО_БАНКІВСЬКОГО_ОСЕРЕДКУ

СІМ_ВІСІМ_ТРИ_ОДИН_ТРИ_ВІСІМ

ТА_ВІДПОВІДЬ_НА_СЕКРЕТНЕ_ЗАПИТАННЯ

ЦЕ_СЛОВО_ХОЛОДЕЦЬ

Закодоване повідомлення:

ЕЯК_ҐЛҐНҐЦЦС_З_ҐТУЕ_ВЕҐТ__ССО

ФЄЗЄХЄУОФ_Ї_ЇСХШЦ_ОШНЄУЗЙТЧХШЧЄХИҐЙДТХСЇ_Ґ_ХСЗ_ЖИЬ

ЧЧМ_ННН_ШЧТТФЦН__ІМТЖШМ__НЦУЖ

ЧУДЧЛДЄ_ТЧ_МЦІДЄИІ_ТМВПКТИ_ХДҐФТІФ

ЩО_САЗСШЖ__ДСЗ_ФСОЩ

Знаючи наперед всі кроки шифрування, повідомлення легко розшифровується.Одержувач знає, що спочатку потрібно визначити параметри матриціm*nпершого рядка. Параметрm легко визначити, порахувавши кількість рядків у листі, потім за допомогою формули $[(L_1-1)/m] +1$визначаємо кількість стовпців матриці першого рядка. Записуємо по стовпцях і зчитуємо по рядках, в отриманому реченні порахувавши кількість слів - визначаємо ключ зсуву і розшифруємо рядок.

1.3.6 Програмна реалізація

Алгоритм шифрування Цезаря полягає заміні кожного символу вхідного повідомлення на символ, який знаходиться на деякій константній відстані з правої чи лівої сторони. Відстань при цьому називають – ключем.

Наприклад для ключа 5 отримаємо послідовність:

Алфавіт:

А Б В Г Ґ Д Е Є Ж З И І Ї Й К Л М Н О П Р С Т У Ф Х Ц Ч Ш Щ Ь Ю Я

Шифр:

Д Е Є Ж З И І Ї Й К Л М Н О П Р С Т У Ф Х Ц Ч Ш Щ Ь Ю Я А Б В Г Ґ

Тобто А замінюємо на Д, Б на Е, і т. д.

Математично шифр Цезаря можна описати наступними формулами:

Encrypt(mn) = (Q + mn + k) % Q;

Decrypt(cn) = (Q + cn - k) % Q.

де m - відкритий текст, k - ключ шифрування, Q - кількість символів в алфавіті, c - зашифрований текст.

Реалізація шифрування Цезаря.

```
using System;

public class CaesarCipher
{
    //символи української абетки
    const string alfabet = "АБВГҐДЕЄЖЗИІЇЙКЛМНОПРСТУФХЦЧШЩЬЮЯ";

    private string CodeEncode(string text, int k)
    {
        //додаємо до алфавіту малі літери
        var fullAlfabet = alfabet + alfabet.ToLower();
        var letterQty = fullAlfabet.Length;
        var retVal = "";
        for (int i = 0; i < text.Length; i++)
        {
            var c = text[i];
            var index = fullAlfabet.IndexOf(c);
            if (index < 0)
            {
                //якщо літеру не знайдено, додаємо її незмінною
                retVal += c.ToString();
            }
            else
            {
                var codeIndex = (letterQty + index + k) % letterQty;
                retVal += fullAlfabet[codeIndex];
            }
        }
```

```
        return retVal;
    }

    //шифрування тексту
    public string Encrypt(string plainMessage, int key)
        => CodeEncode(plainMessage, key);

    //дешифрування тексту
    public string Decrypt(string encryptedMessage, int key)
        => CodeEncode(encryptedMessage, -key);
}

class Program
{
    static void Main(string[] args)
    {
        var cipher = new CaesarCipher();
        Console.Write("Введіть текст повідомлення: ");
        var message = Console.ReadLine();
        Console.Write("Введіть ключ: ");
        var secretKey = Convert.ToInt32(Console.ReadLine());
        var encryptedText = cipher.Encrypt(message, secretKey);
        Console.WriteLine("Зашифроване повідомлення: {0}", encryptedText);
        Console.WriteLine("Розшифроване повідомлення: {0}", cipher.Decrypt(encryptedText,
        Console.ReadLine();
    }
}
```

Рисунок 1.8 Реалізація шифрування Цезаря

Результат роботи програми:

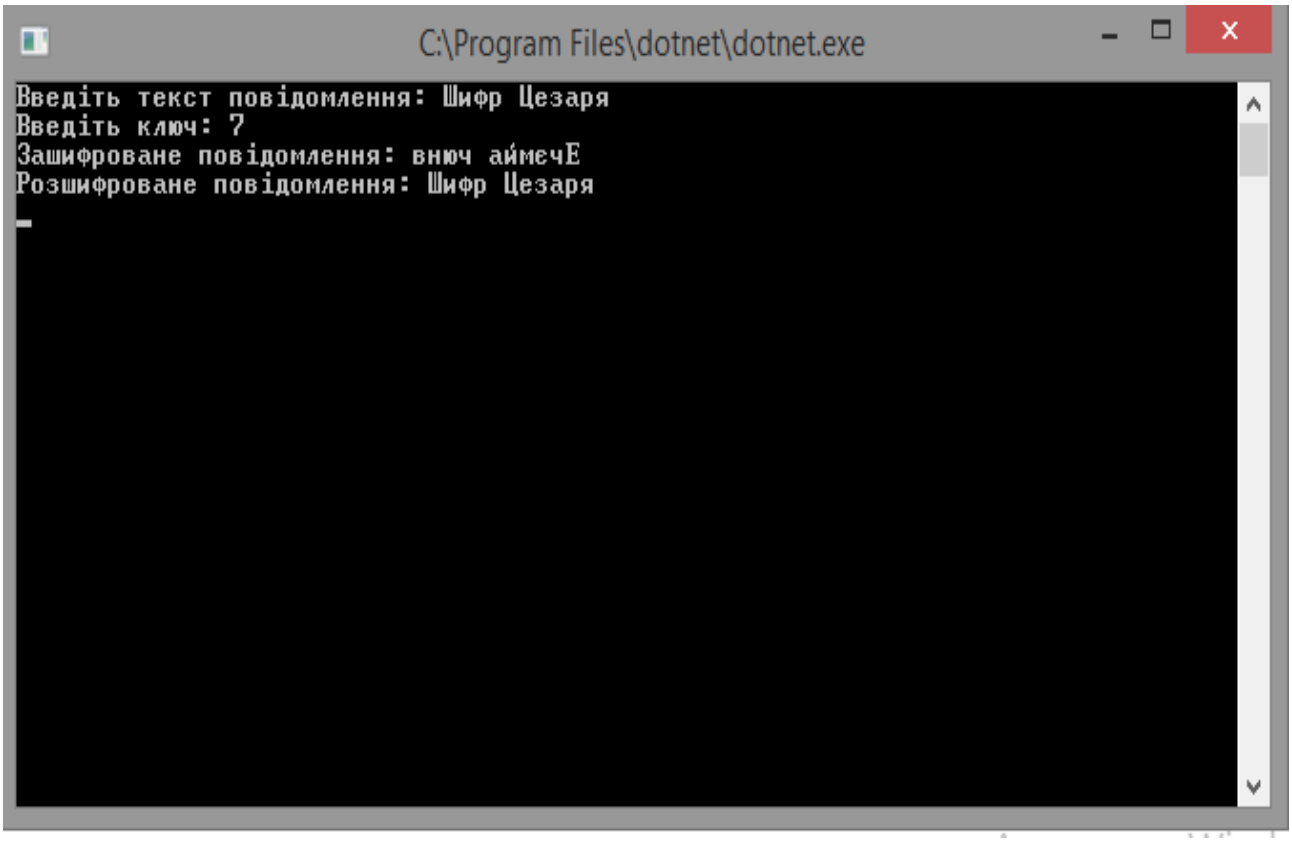


Рисунок 1.9 Результат роботи програмишифрування Цезаря

Шифр Цезаря – один з найдавніших задокументованих шифрів, але оскільки існує всього варіантів ключа, що в середньому складає від 25 до 35 символів, то шифр можна зламати шляхом повного перебору усіх можливих ключів.

Як приклад можна навести такий спосіб: на аркуші у стовпчик виписується увесь алфавіт, кількість аркушів рівна кількості символів у повідомлені (насправді, їх кількість можна зменшити до ―відстані унікальності‖). Після цього стрічки складаються поруч, щоб одна з горизонтальних ліній літер утворили зашифроване повідомлення. Достатньо прочитати усі інші утворені варіанти, і якщо довжина розглянутого шифротексту більше відстані унікальності, то можна буде однозначно визначити зміст шифротексту (оскільки змістовний варіант буде єдиним). Відстань від шифротексту до відкритого тексту буде ключем шифру (за модулем довжини алфавіту).

Шифр Скитала – це шифрування тексту за допомогою дерев'яного циліндру та пергаменту, також відомий як шифр Давньої Спарти. Цей метод шифрування використовувався античними спартанцями та греками, для обміну повідомленнями під час війни.

Шифрування повідомлення з ключем 4, можна представити у вигляді таблиці, де відкритий текст записується в рядки, а розмотування смужки – це склейка всіх стовпців в один:

Таблиця 8. Таблиця шифру Скитала

Ш	И	Ф	Р	_
Д	А	В	Н	Ь
О	Ї	_	С	П
А	Р	Т	И	_

Виходить словосполучення "ШИФР ДАВНЬОЇ СПАРТИ" перетворюється в "ШДОАИАЇРФВ_ТРНСИ_ЬП".

Реалізація шифру Скитала

```
using System;

public class ScytaleCipher
{
    public string Encrypt(string text, int d)
    {
        var k = text.Length % d;
        if (k > 0)
        {
            //доповнюємо рядок пробілами
            text += new string(' ', d - k);
        }

        var column = text.Length / d;
        var result = "";

        for (int i = 0; i < column; i++)
        {
            for (int j = 0; j < d; j++)
            {
                result += text[i + column * j].ToString();
            }
        }

        return result;
    }

    public string Decrypt(string text, int d)
```

```
        var column = text.Length / d;
        var symbols = new char[text.Length];
        int index = 0;
        for (int i = 0; i < column; i++)
        {
            for (int j = 0; j < d; j++)
            {
                symbols[i + column * j] = text[index];
                index++;
            }
        }

        return string.Join("", symbols);
    }
}

class Program
{
    static void Main(string[] args)
    {
        var scytale = new ScytaleCipher();
        Console.Write("Введіть текст повідомлення: ");
        var message = Console.ReadLine();
        Console.Write("Введіть діаметр циліндра: ");
        var diameter = Convert.ToInt32(Console.ReadLine());
        var encText = scytale.Encrypt(message, diameter);
        Console.WriteLine("Зашифрований текст: {0}", encText);
        Console.WriteLine("Розшифрований текст: {0}", scytale.Decrypt(encText, diameter));
        Console.ReadLine();
```

Рисунок 1.10 Реалізація шифру Скитала

Результат роботи програми:

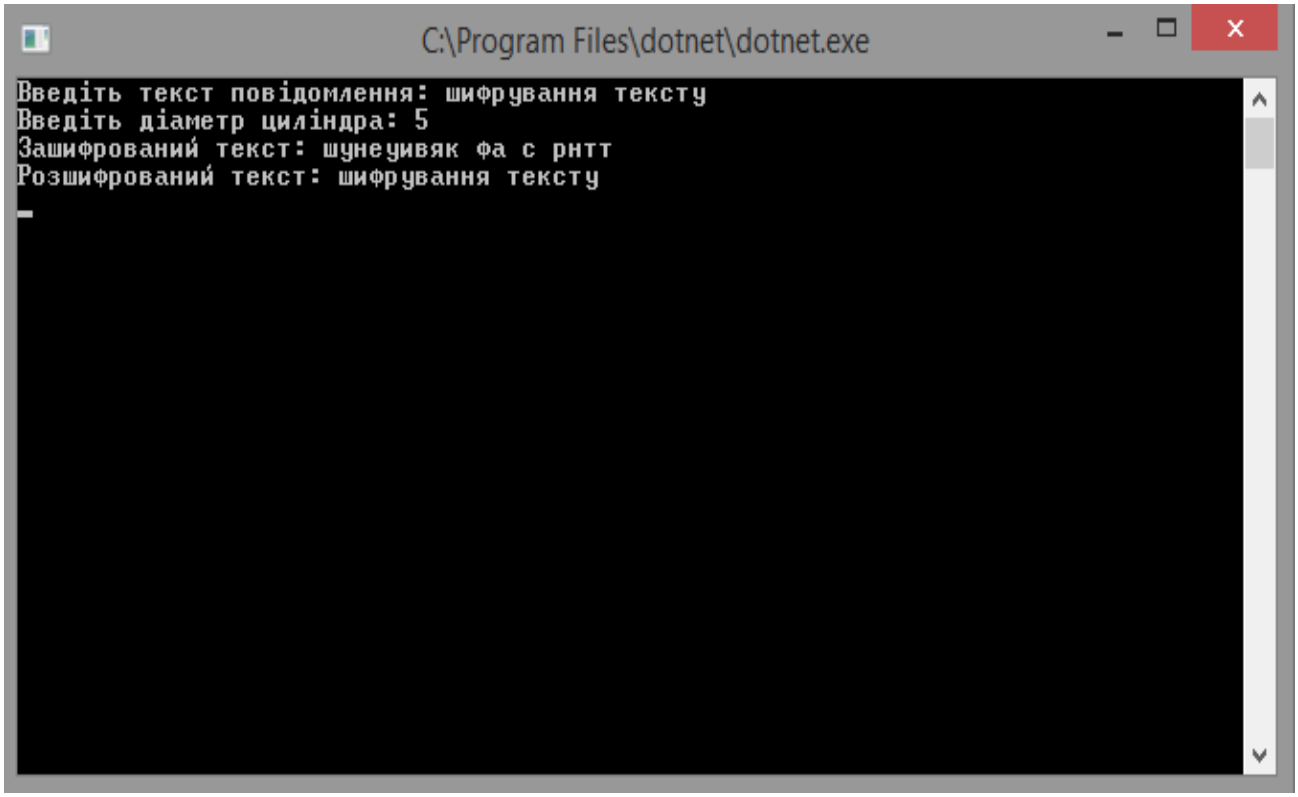


Рисунок 1.11 Результат роботи програми шифрування Скитала

Перевага алгоритма шифрування Скитала у простоті та відсутності помилок. Однак він може бути легко дешифрований. Суть методу в тому, що не знаючи точного діаметра палички, можна використовувати конус, що має змінний діаметр і переміщати пергамент з повідомленням за його довжиною доти, доки текст не почне читатись – таким чином дешифрується ключ – діаметр скитали.У нашому варіанті його застосування також можна змінювати розміри матриці і натрапити на правильний, якщо звичайно знати заздалегідь, що застосовується саме цей метод шифрування.

Спільне застосування цих двох старовинних шифрів, дають велику перевагу в порівнянні з індивідуальним використанням, а придуманий мною метод шифрування параметрів шифрів у повідомленні підвищує криптостійкість всього алгоритму.

Звичайно, кожен повинен вибирати той спосіб захисту свого конфіденційного листування, який особисто йому здається найбільш зручним, але знання ще одного варіанта ніколи не буває зайвим.

Аналіз та програмна реалізація шифру

Щоброзшифруватиповідомлення,потрібнолишезнатисам алгоритм шифрування.Той,хтознаєалгоритмшифрування,можелегкорозшифрувати секретне повідомлення.Отже,ключемуцьомуметоді є сам алгоритм.

Модифікуємо шифр Цезаря для його вдосконалення. Можна було б спробувати розширити алфавіт з 33 до 36 символів і більше за рахунок включення розділових знаків, апострофа і пробілів. Це збільшення алфавіту замаскувало б довжину кожного окремого слова.

Припустимо, що букви зсуваються не на три знака вправо, а на n (0 <n <33). В цьому випадку в системі шифрування з'являється ключ - число n - параметр зсуву. Так як n може приймати різні значення, знання одного тільки алгоритму не дозволить противнику розшифрувати секретне повідомлення.

Зловмисник,якийнамагаєтьсядізнатисязначення секретного ключа,можеатакуватизашифрованийтекстлишешляхомпослідовного перебору всіх можливих ключів (так званий методгрубої сили).При такому методі відбувається пошук осмисленого повідомлення. Таке допущення про одиничність рішення цілком обґрунтовано, коли вихідне повідомлення складено на одній з природних мов і містить більше п'яти-шести знаків. Але якщо повідомлення дуже коротке, можливих рішень може бути кілька. Єдине рішення також дуже важко знай- ти, якщо вихідне повідомлення, складається, наприклад, з цифр.

При методі послідовного перебору всі отримані варіанти будуть рівнозначні і зловмисник не зможе зрозуміти, яка саме комбінація істинна. Аналізуючи шифротекст, він не зможе знайти значення секретного ключа.

Звичайно, у якийсь момент один з варіантів підійде до кода, але в настільки простий метод шифрування не можна розраховувати на більшу секретність.

На основі цих міркувань була побудована математична модельмодифікованого шифра Цезаря. Якщо співставити кожному символу алфавіту його порядковий номер (проводячи нумерацію з 0), то шифрування і дешифрування можна висловити формулами модульної арифметики:

$y = (x + k) \bmod n; x = (y – k + n) \bmod \ldots n,$

Програмну реалізація шифру вирішено робити на мові програмування C++.

Програмна реалізація шифрування / дешифрування приведено в Додатку А.

2. ЕКОНОМІЧНА ЧАСТИНА

2.1 Резюме

Темою даного дипломного проектує розробка та реалізація алгоритму шифрування e-mailповідомлень.Розроблений алгоритм шифрування є результатом спільного використання шифрування Цезаря та алгоритму шифрування Скитала. Спільне застосування цих двох шифрів дають велику перевагу в порівнянні з індивідуальним використанням і підвищує криптостійкість всього алгоритму.

Ефективність програмного продукту визначається його якістю та ефективністю процесу розробки. Якість ПП визначається наступними складовими: з точки зору користувача; з позиції використання ресурсів; виконання вимог до програмного забезпечення.Оцінка якості програмного продукту включає визначення трудомісткості і вартості його створення.

Проведемо розрахунки визначення трудомісткості розробки даного програмного продукту.

2.2. Визначення трудомісткості розробки програмного забезпечення.

Тривалість розробки програмного продукту залежить від його обсягу, трудомісткості розробки, кваліфікації виконавців, а також планових термінів, визначених умовами ринку.

У таблиці 2.1 представлені аналоги програмного забезпечення, функції яких, у більшому або меншому ступені, виконує розроблений програмний продукт.

Таблиця 2.1- Каталог аналогів

Найменування ПП	Обсяг функції ПП – V_o, усл. машинних командах.
1. ПП СУБД	2500 – 9800
2. Комплексні системи ведення БД	950 – 7430
3. ПП введення інформації	1060 – 5750

Методом структурної аналогії по відповідних каталогах аналогів програмного забезпечення визначаємо обсяг програмних засобів, у тисячах умовних машинних команд програми аналога.Для нашого варіанта виділено сірим кольором.

Вибравши аналог ПП, що містить Vo в умовних машинних командах, трудомісткості визначаємо на основі табл.2.2

Таблиця.2.2

Обсяг ПП, тис.умов.машинних команд	Норма часу, люд/год
1.00	229
2.00	244
3.00	262
4.00	283

На підставі отриманого значення, по довіднику, визначаємо укрупнену норму часу на розробку аналога програмного забезпечення (коректується поправочним коефіцієнтом враховуючої умови розробки ПП, тобто в умовах комп'ютера, Кк=0,7÷0,8): Тар = 229 х 0,8 = 183,2 (люд/годин).

Трудомісткість програмного продукту визначається по кожномуетапу розробки окремо на підставі трудомісткості аналога з урахуванням складності розробки, ступеня новизни і ступеня використання в розробці стандартних модулів на підставі формул:

$$T_{ТЗ} = T^a p \times L_1 \times K_H \qquad (2.1)$$

$$T_{ТП} = T^a p \times L_2 \times K_H \qquad (2.2)$$

$$T_{РП} = T^a p \times L_3 \times K_H \times K_T \qquad (2.3)$$

Для розрахунку необхідні наступні коефіцієнти:

Li – питома вага і-го етапу розробки (див. табл. 2.3.);

Кн – поправочний коефіцієнт, що враховує ступінь новизни (див. табл. 2.4.);

Кт – поправочний коефіцієнт, що враховує ступінь використання в розробці типових програм (див. табл. 2.5.).

Таблиця 2.3 Значення питомих коефіцієнтів трудомісткості і-го етапу в загальній трудомісткості розробки ПП.

Код стадії	Ступінь новизни		
	А	Б	В
ТЗ (L_1)	0,15	0,12	0,12
ТП (L_2)	0,16	0,15	0,11
РП (L_3)	0,55	0,58	0,61

Для нашого варіанта виділено сірим кольором.

Таблиця 2.4 - Значення поправочного коефіцієнта, що враховує ступінь новизни

Код ступеня новизни	Ступінь новизни	Значення $К_н$
А	Принципово нові ПО	1,75 – 1,2
Б	ПО – розвиток визначеного параметричного ряду	1,0 – 0,8
В	ПО маючий аналог	0,7

Для нашого варіанта виділено сірим кольором.

Таблиця 2.5. Значення коефіцієнта ступеня використання в розробці типових програм

Ступінь охоплення реалізованих функцій розроблювального ПО типовими програмами, %	Значення $К_т$
60 і вище	0,6
40-60	0,7
20-40	0,8
До 20	0,9

Для нашого варіанта виділено сірим кольором.

Тепер розраховуємо трудомісткість по кожному етапу окремо:

Трудомісткість технічного завдання

$T_{тз}=Ta*L_1*K_н=183,2 *0,12*0,7= 15,39$ (люд/годин) (2.1)

Трудомісткість розробки технічного проекту

$T_{тп}=Ta*L_2*K_н=183.2 *0,11*0,7 = 17,42$ (люд/годин) (2.2)

Трудомісткість розробки робочого проекту

$T_{рп}=Ta*L_3*K_н*K_т=183,2 *0,61*0,7*0,7=54,76$ (люд/годин) (2.3)

Для подальших розрахунків визначили кількість папера, витраченого на кожен етап:технічне завдання$N_{тз}$=2 (стр), розробка ТП $N_{тп}$=18(стр), розробка робочого проекту$N_{рп}$=20 (стр), пояснювальна записка відповідно$N_{пз}$=10 (стр)

Розрахунок зведений у таблицю 2.6

Таблиця 2.6.Розрахунок трудомісткості ПП

Найменування етапів	Розрахунок, годин.		
1.ТЗ	$T_{Ртз}=15,39$	$T_{кк}=0,7*N_{тз}=$ 0,7*2=1,4	$T_{нк}=0,15*N_{тз}=0,15*$ 2=0,30
2.Розробка ТП	$T_{Ртп}=14,12$	$T_{кк}=0,7*N_{тп}=0,7*18=$ 19,6	$T_{нк}=0,15*N_{тп}=0,15*$ 18=4,2
3.Розробка РП	$T_{Ррп}=54,76$	$T_{кк}=0,7*N_{рп}=0,7*20=$ 25,9	$T_{нк}=0,15*N_{рп}=0,15*$ 20=5,55
4.Розробка ПЗ	$T_{пз}=1,5**N_{пз}=$ 1,5*30 =45	$T_{кк}=0,7*N_{тз}=0,7*10=$ 21	$T_{нк}=0,15*N_{пз}=0,15*$ 10 =4,5
Усього, в т.ч.:	231,56		
- на розробку	$\Sigma T_р=149,11$		
- контроль		$\Sigma T_{кк}=67,8$	
-нормоконтроль			$\Sigma T_{нк}=14,55$

2.3 Розрахунок ціни програмного продукту.

У цьому розділі для визначення ціни розраховуємо основну заробітну плату виконавців, матеріальні витрати, вартість машино – години і витрати на розробку ПО. Розрахунок основної заробітної плати виконавців приведений у таблиці 2.7. Відповідно до статті 8 «Закону про Державний бюджет України на 2023» встановлено мінімальну заробітну плату у місячному розмірі з 1 січня 2023 року - 6700 гривень; мінімальну погодинну тарифну ставку – 40.46 грн.

Таблиця 2.7. Розрахунок основної заробітної плати виконавців

Найменування робіт	Трудомісткість робіт, години	Погодинна тарифна ставка, грн.	Розрахунок, грн.
1.Розробка ПП	149,11	40,46	6032,99
2.Контроль керівника	67,8	70,00	4746,00
3.Нормоконтроль	14,55	70,00	1018,15
Усього	-	-	ΣЗо= 11797,49

Зробимо розрахунок матеріальних витрат на розробку ПП. Розрахунок зведемо в таблицю 2.8

Таблиця 2.8- Розрахунок матеріальних витрат на розробку ПЗ

Найменування матеріальних вит.	Тип, модель	Кількість	Ціна одиниці, грн.	Вартість, грн.
Папір	Лист А4	52	4.00	208,0
Транспортно – заготівельні Витрати (10%)				$Втр\_з = 0{,}1 \times Вм1 =$ 0,1х208=21,0
Усього				$В_м = В_{мі} + В_{тр\_з=}$ 229.0

На підставі отриманих даних по окремих статтях витрат складена калькуляція планової собівартості в цілому ПП за формою, приведеною в таблиці 2.9.

Таблиця 2.9.- Розрахунок статей витрат планової собівартості

Стаття витрат	Значення, грн.	Формула розрахунку
1. Матеріали	229.0	$В_м$(див. табл. 2.8)
2. Основна заробітна плата	11797,49	$З_о$ (див. табл. 2.7)
3.Додаткова заробітна плата	1179,75	$Зд = 0{,}1 \times Зо$ =11797,49х0,1
4.Відрахування до єдиногофонду соціального внеску	2854,99	$Вє.с.в. = 0{,}22 \times (Зо + Зд)$ =0,22х (11797,49+1179,75)
5. Накладні витрати	4718,99	$Внак. = 0{,}4 \times Зо$ =0,4 х 11797,49
6. Повна собівартість	25499,22	$С_{пов}$=$В_м$+$З_о$+ $З_д$+Вє.с.в. + Внак.= 229,00+11797,49+1179,75+2854,99+ 4718,99

Розмір прибутку, що включається в ціну, визначаємо по наступній формулі:

$П=(С_{пов}*Р)/100= (25499,22*10)/100 = 2549,92$грн (2.4)

Де р – плановий рівень рентабельності (10-20%).

Оптова ціна (кошторисна вартість) визначається по формулі:

$Ц_о=С_п+П= 25499,22+2549,92 = 28049,14$ грн (2.5)

Виходячи з отриманих даних, ціна реалізації розробленого програмного продукту на основі наступної формули, становитиме:

$Ц_р=Ц_о+ПДВ = 28049,14 + 28049,14*0,2= 33658,96$ грн (2.6)

3 ОХОРОНА ПРАЦІ

Вирішення завдань охорони праці базується на досягненнях ергономіки, наукової організації праці, технічної естетики, гігієни та фізіології праці, психофізіології. Крім того, успіх охорони праці визначається темпами впровадження передової техніки, підвищення рівня механізації і автоматизації виробничих процесів, удосконаленням технології та організації виробництв.

Безпека праці на підприємстві може бути на належному рівні тільки тоді, коли всебічно відповідає вимогам трудового законодавства, державним стандартам України, норм і правил, розроблених для збереження здоров'я працюючих. Важливе місце при цьому належить виконанню організаційних вимог з охорони праці, а також трудовій та виробничій дисципліні працюючих.

Дипломним проектом розглядається питання розробки та реалізації алгоритму шифрування e-mail повідомлень, що передбачає працю з використанням персонального комп'ютера. Тому до розгляду в даному розділі беремо працю програміста та питання забезпечення його здорових умов праці.

1.Аналіз небезпечних і шкідливих факторів, що впливають на програміста при розробці даного програмного комплексу

На робочому місці розробника програмного забезпечення: підвищений рівень отримуваного електромагнітного випромінювання, статична електрика, високий рівень шуму, несприятливі умови мікроклімату, підвищена напруга на зір та мозок тощо.

Робота з ПК відноситься до розумового виду праці з переважно нерухомим видом виконання. Це при постійному впливі призводить до професійних захворювання зору, рухової системи, нервової системи, перевтомлення.

2 Гігієнічні вимоги до виробничого середовища.

Робота з комп'ютером характеризується значною розумовою напругою і нервово-емоційним навантаженням операторів, високою напруженістю зорової

роботи і достатньо великим навантаженням на м'язи рук при роботі з клавіатурою

ЕОМ. В процесі роботи з комп'ютером необхідно дотримувати правильний режим праці і відпочинку.

2.1 Вимоги до приміщення

Розміщення робочих місць з ВДТ ЕОМ і ПЕОМ у підвальних приміщеннях, на цокольних поверхах заборонено. Площа на одне робоче місце становить не менше 6,0 м^2, а об’єм – не менше ніж 20,0 м^3. У приміщеннях слід щоденно робити вологе прибирання. Вони повинні бути оснащені аптечками першої медичної допомоги. При приміщеннях мають бути обладнанні побутові приміщення для відпочинку.

Обємно-планувальні рішення будівель та приміщень для роботи з ВДТ мають відповідати вимогам ДСанПІН 3.3.2.007-98.

2.2 Освітлення

Приміщення для роботи з ВДТ повинні мати природне та штучне освітлення, відповідно до ДБН В.2.5-28-2018. У приміщеннях доцільно, щоб вікна були орієнтовані на північ або північний захід. На вікнах повинні бути штора або жалюзі, що регулюють рівень освітленості і захищають від прямого влучення сонячних променів на робоче місце. При кольоровому оформленні виробничих і допоміжних приміщень необхідно враховувати орієнтацію їхніх вікон стосовно частин світу і використовувати гармонійне сполучення кольорів. Для стін і робочих поверхонь використовують мало насичені (основні) кольори, для невеликих помешкань або ділянок, що рідко потрапляють у поле зору працюючих, а також для створення контрастності – кольори середньої насиченості (допоміжні), для маленьких по площі поверхонь – насичені (акценти) – як функціональне фарбування. Стелі у всіх приміщеннях повинні бути білими. Поверхні устаткування в приміщеннях повинні бути матовими або напівматовими, для виключення випадку відблисків світла в очі працюючого, а стіни бути пофарбованими фарбами пастельних тонів.

2.3 Шум

Оптимальні показники рівня шумів у робочих приміщеннях конструкторськихбюро, кабінетах розраховувачів, програмістів визначаються за ГОСТ 12.1.003-83.Припустимий рівень шуму при розумовій праці, що вимагає зосередженості для програміста, - 50 дБ. Для зменшення шуму й вібрації в приміщенні устаткування, апарати й прилади встановлюються на спеціальні фундаменти й прокладки, що амортизують. Якщо стіни й стелі приміщення є джерелами шумо-утворення, вони повинні бути облицьовані звуковбирним матеріалом.

У виробничих приміщеннях на робочих місцях мають забезпечуватись оптимальні значення повинні бути облицьовані звуковбирним матеріалом.

2.4Мікроклімат

У виробничих приміщеннях на робочих місцях мають забезпечуватись оптимальні значення параметрів мікроклімату: температури, відносної вологості й рухливості повітря – ДСН 3.3.6.042 – 99 Санітарні норми мікроклімату виробничих приміщень..

Таблиця 5.1

Параметри мікроклімату	значення параметрі	
	Взимку	влітку
Температура, C^0	22-24	23-25
Відносна вологість, %	40-60	40-60
Швидкість руху повітря, м/с	0,1	0,1-0,2

Для підтримки в приміщеннях нормального, що відповідає гігієнічним вимогам складу повітря, видалення з нього шкідливих газів, пилу використовують вентиляцію. Механічна вентиляція (кондиціонери, вентилятори і т.д.) залежно від напрямку руху повітряних потоків, може бути витяжною, припливною і припливно-витяжною. При природній вентиляції (за допомогою вікон) повітря надходить у приміщення і видаляється з нього внаслідок різниці температур і тиску.. Механічна вентиляція забезпечується

вентиляторами, що забирають повітря зовні і направляє його до будь-якого робочого місця. або устаткування, а також видаляють забруднене повітря

2.4 Вимоги до організаціїробочого місця працівника

Велике значення має раціональна конструкція та розташування елементів робочого місця, що важливо для підтримки оптимального робочого стану для працівника.

Робочі місця повинні бути розташовані так, щоб у поле зору працюючого не попадали поверхні, що мають властивість віддзеркалювання, вікна освітлювальні прилади. Відеотермінали повинні встановлюватися під кутом 90-100 градусів від вікон, так, щоб світло падало з боку. Робочі місця з ВДТ доцільно розміщати в глибині приміщення. Розташування відео термінала, при якому працюючий звернений обличчям або спиною до вікон, неприпустимо при будь-якому способі реалізації загального висвітлення, як прямим, так і відбитим світлом.

Робочий стіл повинен регулюватися по висоті в границях 680-800 мм, а ширина – забезпечувати можливість виконання операцій в зоні досяжності моторного поля Рекомендовані розміри столу: висота 725 мм, ширина 600-1400 мм, глибина 800-1000 мм. Робочий стілець повинен бути оснащений підйомно-поворотним пристроєм для регулювання висоти сидіння і спинки, а також кута її нахилу. Регулювання кожного параметра повинне вироблятися легко, бути незалежним і надійно фіксуватися.

Розташування екрана ВДТ має забезпечувати зручність зорового спостереження у вертикальній площині під кутом $+30^0$ до нормальної лінії погляду працюючого.

Клавіатуру слід розташовувати на поверхні столу на відстані 100…300 мм від краю, звернутого до працюючого.

3 Пожежна безпека

Пожежнабезпека пристрою, щопроектується у даному дипломному проекті, маєзабезпечуватися відповідно до ГОСТ 12.1.004-91 «Пожарная

безопасность. Общие требования», а вибуховабезпека -- у відповідності до ГОСТ 12.1.010-76 «Взрывобезопасность. Общие требования».

Пожежна безпека -- це такий стан об'єкта, при якому з регламентованою ймовірністю виключається можливість виникнення й розвитку пожежі та впливу на людей небезпечних факторів пожежі, а також забезпечується захист матеріальних цінностей.

Системи запобіганн япожеж, а також протипожежного захисту у сукупності повинні виключати вплив на людей небезпечних факторів пожежі

Пожежна безпека об’єкта забезпечується:

- Системою запобігання пожежі;
- Системою протипожежного захисту;
- Організаційно-технічними заходами.

Всі приміщення повинні бути забезпечені первинними засобами пожежогасіння: пожежним водопостачанням (пожежні крани ПК), пожежні щити з набором пожежного інструменту, вуглекислотними або порошковими вогнегасниками.

У випадку виникнення пожежі необхідно відключити електроживлення, викликати по телефону 101 пожежну команду, евакуювати людей із приміщення відповідно до плану евакуації і приступити до ліквідації пожежі.

ВИСНОВКИ

У дипломній роботі розроблено принципово новий підхід до шифрування повідомлень, яким може скористатися кожен. На мій погляд шифр що оптимально поєднує зручність і надійність збереження наших даних у секреті.

Одна з переваг цього алгоритму це кодування ключів дешифрування безпосередньо в самому повідомленні. Це підвищує стійкість алгоритму до зламів і унеможливлює перехоплення ключів дешифрування.

Розроблений алгоритм шифрування є результатом спільного використання шифрування Цезаря та алгоритма шифрування Скитала.

Спільне застосування цих двох старовинних шифрів, дають велику перевагу в порівнянні з індивідуальним використанням, а придуманий мною метод шифрування параметрів шифрів у повідомленні підвищує криптостійкість всього алгоритму.

Незважаючи на важливість шифрування електронної пошти, використання шифрування не є універсальним та може мати свої обмеження. Наприклад, в деяких випадках шифрування може затримати доставку повідомлення або зробити його менш доступним для пошукових систем.

Крім того, шифрування не є повністю ефективним без використання сильних паролів та інших методів захисту. Найкраще рішення - це комбінація шифрування та інших заходів захисту, таких як використання надійних паролів та захист від шпигунського ПЗ.

Узагалі, шифрування електронної пошти - це важливий аспект безпеки в Інтернеті, який допомагає забезпечити конфіденційність та приватністьнаших повідомлень.

Гібридне шифрування електронної пошти - це метод захисту конфіденційності інформації в електронній пошті шляхом поєднання двох різних методів шифрування. Основною перевагою гібридного шифрування є те, що воно поєднує переваги обох методів шифрування.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Панасенко С. Алгоритми шифрування. Спеціальний довідник. - СПб.: БХВ- Петербург, 2009. - 576 с.
2. Бабенко Л. К., Сидоров І. Д. Паралельні алгоритми криптоаналізу асиметричних систем. Актуальні аспекти захисту інформації в Південному федеральному університеті. Монографія. - Таганрог: Изд-во ТТІ ПФУ, 2011. -С. 207-252.
3. Бабенко Л. К., Іщукова Е. А., Сидоров І. Д. Паралельні Алгоритми факторизації для аналізу асиметричних криптосистем // Матеріали міжнародної конференції «Моделювання сталого регіонального розвитку». - Нальчик:Вид-во КБНЦ РАН, 2011. - С. 78-84.
4. М. Гарленд, С. Ле Гранд та Дж. Ніколлз та ін. Досвід паралельних обчислень із CUDA. IEEE Micro, вип. 28, ні. 4, с. 13-27, 2008.
5. Хабр Сообщество IT–специалистов веб–сайт. URL: https://habr.com (дата звернення 01.06.2021).
6. Онацкий А. В., Йона Л.Г. Асимметричные методы шифрования. – Модуль 2 Криптографические методы защиты информации в телекоммуникационных системах и сетях: Учеб. Пособие/ Под ред. Н.В. Захарченко – Одесса: ОНАС им. А.С Попова, 2010 – 148с.
7. Dut Державний університет комунікацій веб–сайт. URL: http://www.dut.edu.ua(дата звернення 15.06.2021).
8. Бабаш А.В., Баранова Е.К. Оперативные методы криптографии. - М.: РГСУ, 2017. - 104 с.
9. ELAKPI Електронний архів наукових та освітніх матеріалів веб–сайт. URL: https://ela.kpi.ua (дата звернення 15.06.2021).
10. Адаменко М.В. Основы классической криптологии. Секреты шифров и кодов.- М.: ДМК Пресс, 2017. - 256 с.

ДОДАТОК А

Програмна реалізація шифрування / дешифрування

```
#include "stdafx.h" #include <iostream> #include <string>
#include <conio.h> #include <stdlib.h> #include <sstream>
#include <fstream> using namespace std;int main()
{int k; // Змінна вибору -шифрування/дешифрування
int shift; //Величина зсуву
string result = ""; // рядок - результат cout<<" Введіть 1 для шифрування та для
розшифрування 2\n"; cin>>k;switch (k) //Якщо k
{ case 1: // Якщо вибрано шифрування
{ cout<<" Введіть значення зсуву дляшифрування \n";
cin>>shift;
if (shift > 26)
shift = shift % 26; // обчислення зсувуcout<<"Read of file...\n";
// читання файлу
string s; // Рядок, зчитаний з файлуifstream in("Test.txt");
getline(in,s);
cout<<"Текст файлу: \n"<<s<<endl;in.close();
cout<<"Читання завершено!\n";cout<<"Шифрування...\n";
for (int i = 0; i < s.length(); i++) {
// Якщо не латиниця
if (((int)(s[i]) < 65)||((int)(s[i]) > 122))result += s[i];
// Якщо буква є рядковою
if (((int)(s[i]) >= 97) && ((int)(s[i]) <= 122))
// Якщо буква після зсуву виходить замежі алфавіту
{ if ((int)(s[i]) + shift > 122)
// Додавання в рядок результатів символаresult += (char)((int)(s[i]) + shift -
26);
// Якщо буква після зсуву виходить замежі алфавіту
else
// Додавання в рядок результатів символаresult += (char)((int)(s[i]) + shift); }

// Якщо буква є прописною
if (((int)(s[i]) >= 65) && ((int)(s[i]) <= 90)){
// Якщо буква після зсуву виходить замежі алфавіту
if ((int)(s[i]) + shift > 90)
// Додавання в рядок результатів символаresult += (char)((int)(s[i]) + shift -
26);
// Якщо буква може бути зсунута в межахалфавіту
else
// Додавання в рядок результатів символаresult += (char)((int)(s[i]) + shift);
} }cout<<"Шифрування завершено!\n";
cout<<"Результат:\n";
cout<<result; //Вивід результатуbreak; }
```

```
 case 2:
 // Якщо вибрано дешифрування
{ cout<<" Введіть значення зсуву длярозшифрування \n";
 cin>>shift;
 if (shift > 26) shift = shift % 26;
 cout<<"Прочитати файл...\n";string s;
 ifstream in("Test.txt");getline(in,s);
 cout<<"Текст файлу: \n"<<s<<endl;in.close();
 cout<<"Читання завершено!\n";cout<<"Розшифровка...\n";
 for (int i = 0; i < s.length(); i++)
 { if (((int)(s[i]) < 65)||((int)(s[i]) > 122))result += s[i];
 if (((int)(s[i]) >= 97) &&
 ((int)(s[i]) <= 122))
 { if ((int)(s[i]) - shift < 97)
 result += (char)((int)(s[i]) - shift + 26);else
 result += (char)((int)(s[i]) - shift); }if (((int)(s[i]) >= 65) &&
 ((int)(s[i]) <= 90))
 { if ((int)(s[i]) - shift < 65)
 result += (char)((int)(s[i]) - shift + 26);else
 result += (char)((int)(s[i]) - shift);  }       }cout<<"Розшифровка
 завершена!\n"; cout<<"Результат:\n";
 cout<<result; //Вивід результатуbreak; }
 default:
 // Якщо помилкове значення
 { cout<<"Помилка\n";break; } }
   getch(); return 0;  }
```

ДОДАТОК Б

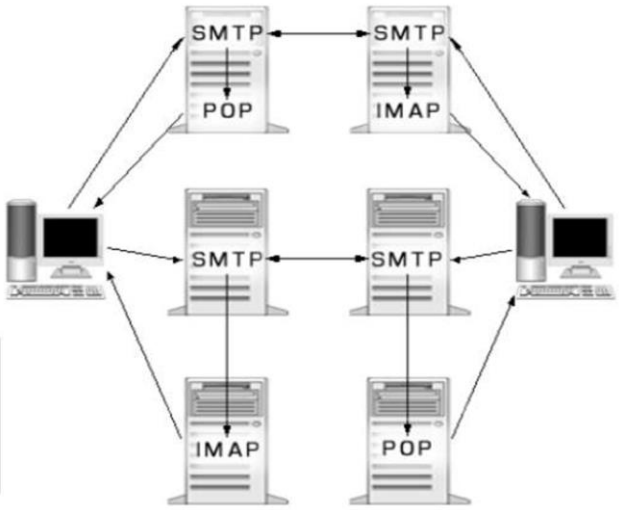
Слайди мультимедійної презентації

Загрози інформаційної безпеки

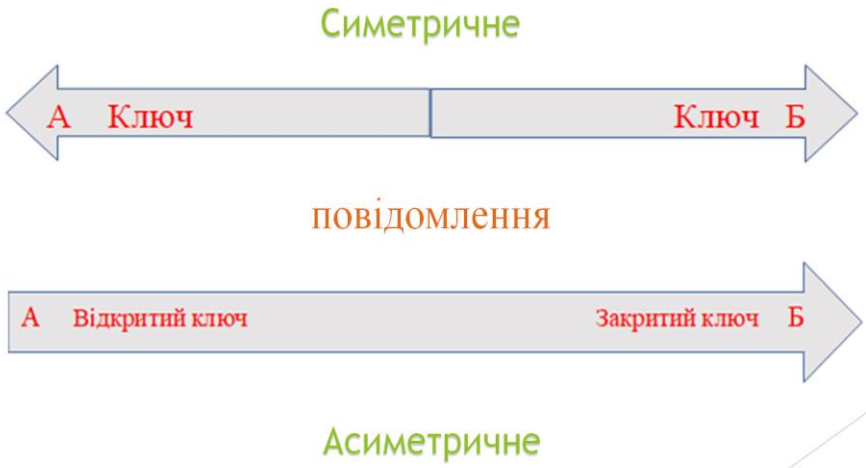
Загрози інформаційної безпеки (через користування ресурсами мережі Інтернет)

- Потрапляння у ІС **шкідливого ПЗ** (вірусіи, троянські програми, мережеві хробаки, клавіатурні шпигуни, рекламні системи і т.д.)
- Інтернет-шахрайство (фішинг)
- **Хакерські атаки** - дії, що спрямовані на порушення штатного режиму функціонування системи, порущення доступності її серверів, отримання несанкціонованого доступу до конфіденційних відомостей, порушення цілісності даних
- Потрапляння до **ботнет-мережі, DDoS-атаки**
- **"Крадіжка особистості"**- несанкціоноване заволодіння персональними дании особи

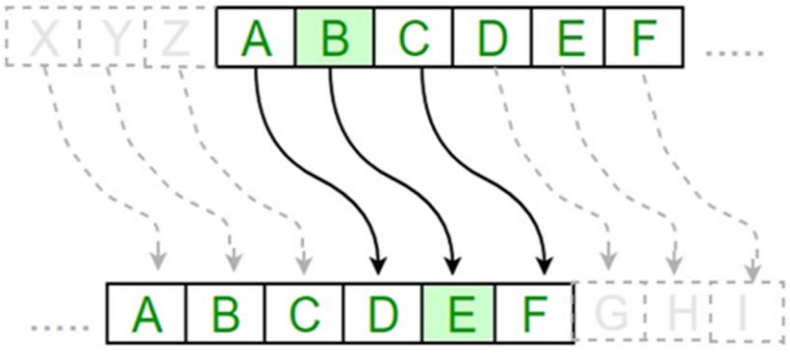
Взаємодія протоколів роботи електронної пошти



Приклад симетричного та асиметричного шифрування



Принцип дії Шифра Цезаря

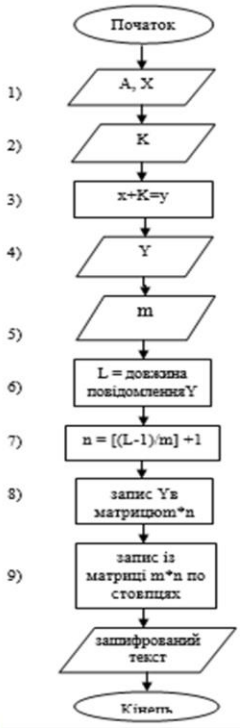


Алгоритм шифрування Скитала

ШИФР ДАВНЬОЇ СПАРТИ

Ш	И	Ф	Р	-
Д	А	В	Н	Ь
О	Ї	-	С	П
А	Р	Т	И	-

ШДОАИАЇРФВ_ТРНСИ_ЬП



Блок-схема алгоритму шифрування та його опис

Шифрування проводиться таким чином:

1) A - визначаємо алфавіт, вводимо відкритий текст X.
2) Вводимо ключ шифрування K (усунення при шифруванні Цезаря).
3) Для отримання символів yi зашифрованого повідомлення (шифром Цезаря), зсуваємо символихі вихідного повідомлення на k позицій праворуч.
4) Визначаємо зашифроване повідомлення Y (шифром Цезаря), яке далі шифруватиметься шифром Скитала.
5) Вводиться "ключ" m - кількість рядків матриці Скитала.
6) Обчислюється L - довжина повідомлення Y.
7) Обчислюється кількість стовпців n за формулою [(L-1)/m]+1
8) Записуємо повідомлення Y в матрицю m*n.
9) Вміст матриці m*n зчитується послідовно стовпцями зверху вниз. Отримуємо результуючий зашифрований текст.

текст повідомлення

1 2 3 4
X1 вітаю шановний пан адвокат

1 2 3 4 5 6 7
X2 надсилаю вам пароль від мого банківського осередку

1 2 3 4 5 6
X3 сім вісім три один три вісім

1 2 3 4 5
X4 та відповідь на секретне запитання

1 2 3
X5 це слово холодець

Отримання символів уі

x+ Ключ 4= y

А Б В Г Ґ Д Е Є Ж З И І Ї Й К Л М Н О П Р С Т У Ф Х Ц Ч Ш Щ Ь Ю Я

X_1 ВІТАЮ_ШАНОВНИЙ_ПАН_АДВОКАТ

Визначаємо зашифроване повідомлення Y

Y_1 ЕЛЦҐВ_ЯҐСТЕСКН_УҐС_ҐЗЕТОҐЦ

Обчислення кількості стовпців n

L_1 – довжина повідомлення Y_1 дорівнює кількості літер у кожному рядку з урахуванням прогалин між словами. У першому рядку нашого повідомлення $L_1=26$.

Обчислюється кількість стовпців n за формулою $[(L_1-1)/m] +1$,

де $[(L_1-1)/m]$ ціла частина числа:

кількість стовпців $n = [(L_1-1)/m]+1= (26-1)/5+1=6$

Отримання зашифрованого тексту першого рядка

Записуємо повідомлення Y_1 в матрицю m*n, дем=5 та n=6.

Y_1 ЕЛЦҐВ_ЯҐСТЕСКН_УҐС_ҐЗЕТОҐЦ

Е	Л	Ц	Ґ	В	_
Я	Ґ	С	Т	Е	С
К	Н	_	У	Ґ	С
_	Ґ	З	Е	Т	О
Ґ	Ц	_	_	_	_

Вміст матриці зчитується послідовно стовпцями зверху вниз. Отримуємо результуючий зашифрований текст (Y_1):

ЕЯК_ҐЛҐНҐЦЦС_З_ҐТУЕ_ВЕҐТ__ССО

Вихідне та закодоване повідомлення

Вихідне повідомлення:

ВІТАЮ_ШАНОВНИЙ_ПАН_АДВОКАТ

НАДСИЛАЮ_ВАМ_ПАРОЛЬ_ВІД_МОГО_БАНКІВСЬКОГО_ОСЕРЕДКУ

СІМ_ВІСІМ_ТРИ_ОДИН_ТРИ_ВІСІМ

ТА_ВІДПОВІДЬ_НА_СЕКРЕТНЕ_ЗАПИТАННЯ

ЦЕ_СЛОВО_ХОЛОДЕЦЬ

Закодоване повідомлення:

ЕЯК_ҐЛҐНҐЦЦС_З_ҐТУЕ_ВЕҐТ__ССО

ФЄЗЄХЄУОФ_Ї_ЇСХШЦ_ОШНЄУЗЙТЧХШЧЄХИҐЙДТХСЇ_Ґ_ХСЗ_ЖИЬ

ЧЧМ_ННН_ШЧТТФЦН__ІМТЖШМ__НЦУЖ

ЧУДЧЛДЄ_ТЧ_МЦІДЄИЇ_ТМВПКТИ_ХДҐФТІФ

ЩО_САЗСШЖ__ДСЗ_ФСОЩ

ВСП «ОДЕСЬКИЙ ТЕХНІЧНИЙ ФАХОВИЙ КОЛЕДЖ ОНТУ»

РЕЦЕНЗІЯ

на дипломний проект (роботу) здобувача (здобувачки) освіти
відділення комп'ютерних систем

Дедігурова Микити Олександровича

(прізвище, ім`я та по батькові)

Спеціальність **123 «Комп'ютерна інженерія»**

Освітня програма **Обслуговування комп'ютерних систем та мереж**

Керівник дипломного проекту (роботи) **Шевцов Юрій Сергійович**

(прізвище, ім`я та по батькові)

Тема дипломного проекту (роботи)

Розробка та реалізація алгоритму шифрування e-mail повідомлень

Обсяг розрахунково-пояснювальної записки 65 сторінок

Обсяг графічної (презентаційної) частини 14 аркушів (слайдів)

ХАРАКТЕРИСТИКА ДИПЛОМНОГО ПРОЕКТУ (РОБОТИ)

а) заключення про ступінь відповідності виконаного дипломного проекту (роботи) завданню

Дипломний проект повністю відповідає завданню до дипломного проектування. Графічна частина складається з окремих слайдів, оформлених у вигляді презентації, передбачених технічним завданням. Якість виконання пояснювальної записки та слайдів добра.

б) характеристика виконання кожного розділу дипломного проекту (роботи)

Пояснювальна записка дипломного проекту виконана якісно, у повному обсязі. В дипломному проекті здобувачем проаналізовано існуючі алгоритми шифрування, та їх сумісне використання. Визначені на основі проведеного аналізу переваги гібридного шифрування. Розроблено новий алгоритм шифрування. В дипломному проекті в останніх розділах проаналізовано економічні показники та питання охорони праці.

в) оцінка якості виконання пояснювальної записки та графічної частини дипломного проекту (роботи) _____

Презентаційні матеріали виконані якісно, демонстративно та відповідають вмісту теоретичного матеріалу

г) перелік позитивних якостей дипломного проекту (роботи)_____

Тематика дипломного проекту є актуальною. Розроблений алгоритм шифрування є результатом спільного використання шифрування Цезаря та алгоритма шифрування Скитала. Спільне застосування цих двох шифрів, дають велику перевагу в порівнянні з індивідуальним використанням. Розроблений шифр може використовуватися при шифруванні e-mail повідомлень.

д) основні недоліки дипломного проекту (роботи) _____

Серед недоліків роботи варто вказати, відсутність посилань на перелік використаних джерел та наявність орфографічних помилок в тексті пояснювальної записки

Оцінка розрахункової частини добре

Оцінка графічної частини добре

Загальна оцінка добре

Прізвище, ім'я, по батькові рецензента **Василіу Євген Вікторович**

Місце роботи і посада рецензента **Державний університет інтелектуальних технологій і зв'язку, д.т.н., проф. кафедри КБ та ТЗІ, декан факультету інформаційних технологій та кібербезпеки**

Підпис: _____

« 16 » 06 2023 р.

ПІДПИС ПОСВІДЧУЮ
НАЧАЛЬНИК ВІДДІЛУ КАДРІВ ДУТ

UNICHECK

Ім'я користувача:
Наталія Вікторівна Копусь

ID перевірки:
1015498625

Дата перевірки:
08.06.2023 10:56:00 EEST

Тип перевірки:
Doc vs Internet + Library

Дата звіту:
08.06.2023 11:03:00 EEST

ID користувача:
100011688

Назва документа: 4КС-56 Дедігуров М.О

Кількість сторінок: 51 Кількість слів: 9693 Кількість символів: 70000 Розмір файлу: 1.43 MB ID файлу: 1015154840

Виявлено модифікації тексту (можуть впливати на відсоток схожості)

45.5%
Схожість

Найбільша схожість: 7.45% з Інтернет-джерелом (https://core.ac.uk/download/pdf/84825478.pdf)

45.5% Джерела з Інтернету | 1000 ..... Сторінка 53

Не знайдено джерел з Бібліотеки

0% Цитат

Вилучення цитат вимкнене

Вилучення списку бібліографічних посилань вимкнене

0%
Вилучень

Немає вилучених джерел

Модифікації

Виявлено модифікації тексту. Детальна інформація доступна в онлайн-звіті.

Замінені символи | 30

Підозріле форматування | 9 сторінок

ВСП «ОДЕСЬКИЙ ТЕХНІЧНИЙ ФАХОВИЙ КОЛЕДЖ ОНТУ»

ВІДГУК

керівника на дипломний проект здобувача (здобувачки) освіти

відділення **комп'ютерних систем**

Дедігурова Микити Олександровича

(прізвище, ім`я та по батькові)

Спеціальність: **123 «Комп'ютерна інженерія»**

Освітня програма: **«Обслуговування комп'ютерних систем та мереж»**

Тема дипломного проекту:

Розробка та реалізація алгоритму шифрування e-mail повідомлень

ХАРАКТЕРИСТИКА ДИПЛОМНОГО ПРОЕКТУ

а) обсяг і якість виконання проекту (графічного матеріалу і розрахунково-пояснювальної записки) **Пояснювальна записка дипломного проекту виконана якісно, у повному обсязі. В дипломному проекті здобувачем проаналізовано існуючі алгоритми шифрування, та їх сумісне використання. Визначені на основі проведеного аналізу переваги гібридного шифрування. Розроблено новий алгоритм шифрування, якій може використовуватися при обміні e-mail повідомленнями. В дипломному проекті в останніх розділах проаналізовано економічні показники та питання охорони праці.**

б) самостійність роботи над проектом: **Здобувач самостійно визначався з напрямом роботи, дослухався до рекомендацій керівника дипломного проекту, пропонував рішення та своєчасно надавав результати роботи, якісно виконував основні етапи роботи за вимогою керівника.**

в) теоретична підготовка випускника (випускниці): **Теоретична підготовка випускника в цілому відповідає існуючим вимогам до фахівців відповідного рівня кваліфікації**

г) вміння розв'язувати виробничі та конструкторські питання **В процесі роботи над дипломним проектом здобувач продемонстрував уміння використовувати останні досягнення науки та техніки в предметній галузі, на підставі відповідної навчальної та науково-технічної літератури, пропонувати технічні рішення поставлених завдань.**

Оцінка розрахункової частини добре

Оцінка графічної частини добре

Загальна оцінка добре

Прізвище, ім'я, по батькові керівника дипломного проекту **Шевцов_Юрій_Сергійович**

Місце роботи і посада керівника дипломного проекту **к.т.н. доцент каф. "Електричної інженерії та електроніки" НУ «Одеська Морська академія»**

Підпис

« **12** » **червня** 20**23** р.

ДОЗВІЛ
НА РОЗМІЩЕННЯ
ВИПУСКНОЇ КВАЛІФІКАЦІЙНОЇ РОБОТИ
В ЕЛЕКТРОННОМУ РЕПОЗИТАРІЇ ВСП «ОТФК ОНТУ»

Ми, що нижче підписалися,

Дедігуров Микита Олександрович,
здобувач освіти гр. 4КС-56, та

Шевцов Юрій Сергійович,
керівник дипломного проекту,

не заперечуємо щодо розміщення електронного варіанту пояснювальної записки до випускної кваліфікаційної роботи молодшого спеціаліста на тему:

«Розробка та реалізація алгоритму шифрування e-mail повідомлень» (автор роботи – Дедігуров М.О., керівник роботи – Шевцов Ю.С.)

виконаного у ВСП «Одеський технічний фаховий коледж Одеського національного технологічного університету» в 2023 році, у повному обсязі в електронному репозитарії ВСП «ОТФК ОНТУ» для вільного доступу через мережу Інтернет.

Несемо відповідальність за ідентичність електронного та друкованого варіантів випускної кваліфікаційної роботи, і даємо згоду на обробку персональних даних.

Виконавець _____ / Дедігуров М.О. /

Керівник _____ / Шевцов Ю.С./

«12» 06 2023 р.