**Лабораторна робота №1**

**Тема роботи:** Симетричне, асиметричне і гібридне шифрування інформації

**Мета роботи:** Ознайомити студентів із принципами роботи симетричного, асиметричного і гібридного шифрування, яке використовується у зашифрованих інтернет протокола.

**Завдання до лабораторної роботи**

Створити програму, яка імітує процес передачі зашифрованих даних у мережі.

Створіть окремий клас Client, що представляє клієнта мережі. У кожного клієнта повинні бути наявними принаймні наступні компоненти:

1. Власна робоча директорія.
2. Відкритий асиметричний ключ шифрування.
3. Закритий асиметричний ключ шифрування.
4. Пароль для симетричного шифрування файлів.
5. Список клієнтів, підключених до поточного клієнту.
6. Методи для передачі інформації іншим клієнтам і отримання інформації від інших клієнтів.

Інформацію, яку передаєте іншим клієнтам, інкапсулюйте в класі Package. Цей клас повинен містити інформацію про відправника, отримувача, і повідомлення або файл. В рамках цієї лабораторної роботи, можете створити два окремих поля – для повідомлення (паролю або звичайного повіпомлення) і файлу (може приймати null, в разі звичайного повідомлення).

Передаваєме повідомлення завжди повинно шифруватися асиметричним алгоритмом, окрім випадків, коли клієнти обмінюються своїми відкритими ключами асиметричного шифрування. Асиметричну генерацію ключів реалізуйте самостійно, без використання можливих готових реалізацій. Самі повідомлення можете шифрувати із використанням бажаних готових реалізацій, за отриманим ключем. Довжину ключа оберіть самостійно. Обґрунтуйте ваш вибір у звіті до лабораторної роботи.

Створіть окремий клас SymmetricFileCypher. Клас повинен містити методи для шифрування та дешифрування файлів за отриманим симетричним ключем. Алгоритм симетричного шифрування оберіть самостійно. Для самого процесу симетричного шифрування файлу, можете використовувати готові реалізації, наприклад із javax.crypto.\*. Опишіть у звіті, який алгоритм і реалізацію обрали.

У основній програмі, створіть мережу із клієнтів – декілька клієнтів, пов’язаних один з одним, щоб їх зв’язок можна було описати у вигляді простого графу (без петель і без циклів, утворених ребрами) і з більше ніж двома вершинами. Спосіб передачі пакетів від одного клієнта до іншого оберіть самостійно. Опишіть мережу і спосіб передачі пакетів між клієнтами у звіті до лабораторної роботи.

Надішліть декілька повідомлень між двома клієнтами, відстань яких один від одного на графі більше одиниці. Клієнти повинні розшифрувати отримані повідомлення своїм приватним ключем асиметричного шифрування і відобразити їх на екрані. Клієнти, які беруть участь у передачі, але не є відправниками або отримувачами, повинні відображати передаваєме повідомлення, але (очевидно) у зашифрованому вигляді.

Передайте файл із завданням до лабораторної роботи (цей документ) між цими двома клієнтами. Файл повинен бути зашифрованим симетричним методом шифрування. Симетричний ключ передається у формі повідомлення, зашифрованого асиметричним ключем шифрування. Отримувач повинен розшифрувати файл у своїй директорії. Клієнти, які беруть участь у передачі, але не є відправниками або отримувачами, повинні зберегти файл, який передають, у себе в директорії у (очевидно) зашифрованому вигляді.

**Контрольні запитання**

1. В чому полягає принцип симетричного шифрування? Які у такого шифрування є переваги та недоліки.

2. В чому полягає принцип асиметричного шифрування? Які у такого шифрування є переваги та недоліки.

3. Перерахуйте декілька відомих вам алгоритмів симетричного і асиметричного шифрування.

4. Опишіть роботу алгоритму Діффі-Геллмана.

5. В чому полягає принципова різниця між алгоритмами Діффі-Геллмана і RSA?

*Відповідь надається у вигляді звіту з лабораторної роботи, до якого потрібно включити інформацію про виконане завдання, обґрунтування обраних алгоритмів та способів їх реалізації, діаграму класів створеної програми, скріншоти виконання програми, відповіді на контрольні запитання, висновки. Окремо додаються файли з вихідним кодом та виконуємі файли із всіма необхідними директоріями та файлами, що знадобились або згенерувались в процесі роботи програми.*