НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ім. Ігоря
СІКОРСЬКОГО»
ФІЗИКО-ТЕХНІЧНИЙ ІНСТИТУТ

Комп'ютерний практикум №3-4

Підготували студенти
групи ФІ-12мн,мп
Кустарьова Катерина
Тулупов Матвій

Київ — 2021

Зміст

1. Безпека електронних платіжних систем

Електронні платіжні системи є одним із найпопулярніших видів роботи з електронною валютою. З кожним роком вони розвиваються все активніше, займаючи досить велику частку ринку роботи з валютою. Разом з ними розвиваються й технології забезпечення їхньої безпеки. Оскільки на сьогоднішній день жодна електронна платіжна система не може існувати без хороших технологій та систем безпеки, які, у свою чергу, забезпечують безпечну транзакцію грошових операцій.

Самих електронних платіжних систем, власне, як і технологій захисту, існує дуже багато. Кожна з них має різні принципи та технології роботи, а також свої переваги та недоліки.

Крім того, залишається невирішеною ціла низка питань теоретичного та практичного характеру, що й визначає актуальність теми дослідження. Кожна електронна платіжна система використовує свої методи, алгоритми шифрування, протоколи передачі для виконання безпечних транзакцій і передачі. Одні системи використовують алгоритм шифрування RSA та протокол передачі HTTPs, інші використовують алгоритм DES та протокол SSL для передачі зашифрованих даних.

2. Криптографічні методи шифрування

Webmoney і «Яндекс.Гроші» застосовується ключ довжиною 1024 біти (дуже високий показник, зламати такий ключ методом простого перебору практично неможливо), а в PayPal використовується ключ вдвічі коротший – 512 біт. Залишається тільки оцінити за цим параметром E-Port. Незважаючи на використання в ній SSL-протоколу і навіть на довжину ключа до 128 бітів, в E-Port є деяка потенційна вразливість: багато старих версій браузерів підтримують шифрування з ключами меншої довжини, тому існує можливість зламати отримані дані; відповідно, для тих, хто використовує браузер як клієнт для платіжної системи, необхідно працювати з його останньою версією (звичайно, це завжди зручно і можливо). Однак у графі «шифрування» для E-Port : система заслужила на високу оцінку завдяки застосуванню прогресивного протоколу PGP для шифрування повідомлень електронної пошти.

3. Платіжна система

Наша програма імітує роботу біржі/банку. Виділили сутності банку та рахунки користувача.

Рахунки бувають різних видів, наприклад, рахунки до запиту та депозити, відповідно буде декілька сутностей рахунків. Насамперед програма використовує Library з набором інтерфейсів, що описує функціональність рахунку. При проектуванні інтерфейсу визначили функціонал без реалізації, та реалізували у класах, що застосовують даний інтерфейс. AccHandler використовується для створення подій. А для обробки подій також визначено клас AccEvent, який визначає дві властивості для читання: повідомлення про подію та сума, на яку змінився рахунок.

Так як такого банківського рахунку немає, а є конкретні рахунки - депозит, до запиту, та інші, то цей клас є абстрактним. У той самий час він реалізує інтерфейс Account.

Кожен рахунок має унікальний ідентифікатор Id – унікальний номер. Для його одержання застосовується статичний counter. Майже всі методи є віртуальними. Метод CallEvent() викликає подію, яка представляє AccHandler та передає йому аргументи події – об'єкт AccEvent.

Для виклику окремих подій за допомогою event передбачені методи Onadd, OnCalculated, OnClosed, OnOpened, OnWith. У методах Put, Withd та Close використовуються метод FindAccount() для отримання рахунку для додавання або виведення коштів, а також закриття.

При закритті рахунку в методі Close() створюється новий масив без одного елемента - рахунку, який треба видалити. Таким чином відбувається видалення рахунку.

У методі CalculatePercentage() пробігаємось по всіх елементах масиву рахунків, збільшуємо у кожного рахунку лічильник днів і робимо нарахування відсотків.

Загальним клас є обгорткою, через яку з головного проекту ми взаємодіятимемо з користувачами. Також слід зазначити, що більшість членів класу мають модифікатор protected internal, тобто їх буде видно лише всередині проекту Library.

Депозитні рахунки мають особливість: вони оформлюються на тривалий період. Наприклад, рахунок має термін 30 днів, і клієнт не може ні

додати кошти, ні зняти їх, але може закрити рахунок. Всі рахунки в класі зберігаються в accounts.

На момент проектування класу ми не знали, якими саме рахунками керуватиме біржа. Можливо, це будуть будь-які рахунки, а можливо лише депозитні, тобто об'єкти DepAcc, тому використовували узагальнення.

Для шифрування даних ми використовуємо стандартну реалізацію RSA. У результаті вийде програма, що імітує роботу банку та взаємодію з користувачем.

4. Висновок

Ми намагалися реалізувати гнучку платіжну систему з використанням вбудованих бібліотек C та засобів ООП. Нам вдалося досягнути поставленої мети та розібрати принцип роботи деяких відомих платіжних ситем.
Ми зробили огляд сучасних бібліотек щифрування, які значно поліпшують роботу для таких задач.