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

Звіт за темою

ДОСЛІДЖЕННЯ ОСНОВНИХ ЗАДАЧ, ЩО ВИНИКАЮТЬ ПРИ ПРОГРАМНІЙ РЕАЛІЗАЦІЇ КРИПТОСИСТЕМ НА ІНТЕЛЕКТУАЛЬНИХ КАРТКАХ ТА ТОКЕНАХ

Виконали студенти
групи ФІ-32мн
Баєвський Констянтин,
Шифрін Денис,
Кріпака Ілля

Київ — 2023

ЗМІСТ

ВСТУП

0.1 Мета практикуму

Дослідити основні задачі, що виникають при програмній реалізації криптосистем. Запропонувати методи вирішення задачі контролю доступу до ключової інформації, що зберігається в оперативній пам'яті ЕОМ для різних (обраних) операційних систем. Запропонувати методи вирішення задачі контролю правильності функціонування програми криптографічної обробки інформації. Порівняти з точки зору вирішення цих задач інтерфейси Crypto API, PKCS11. Дослідити основні задачі, що виникають при програмній реалізації криптосистем. Запропонувати методи вирішення задачі контролю доступу до ключової інформації, що зберігається в оперативній пам'яті ЕОМ для різних (обраних) операційних систем. Запропонувати методи вирішення задачі контролю правильності функціонування програми криптографічної обробки інформації. Порівняти з точки зору вирішення цих задач інтерфейси Crypto API, PKCS11.

0.1.1 Постановка задачі та варіант

Треба виконати	**Зроблено**
Дослідити основні задачі, що виникають при програмній реалізації криптосистем на токенах та сматр картках	✓
Дослідити різницю Crypto API та PKCS11 інтерфейсів	✓
Навести приклад роботи із PKCS11 інтерфейсом із токеном від Yubikey	✓

0.2 Хід роботи/Опис труднощів

На початку роботи над практикума вибрати варіант 1А, та далі продовжували роботу над ними. Згідно вибраного варіанту у даній роботі буде розглянуто методи вирішення на токенах та смарт пристроях. Під час виконання звіту виникала лише одна часова складність.

1 ОСНОВНІ ЗАДАЧІ, ЩО ВИНИКАЮТЬ ПРИ ПРОГРАМНІЙ РЕАЛІЗАЦІЇ КРИПТОСИСТЕМ НА ТОКЕНАХ ТА СМАРТ КАРТКАХ

На початку звіту варто пояснити для чого виокристовуються смарт картки та токени.

1) Смарт-картки – це носії ключової інформації з контактним та/або безконтактним інтерфейсом, що призначені для використання в системах інформаційного доступу, кваліфікованого електронного підпису, документообігу, авторизації користувачів, захищеного зберігання ключової інформації, а також в якості ідентифікаційного інструменту. Тобто, в основному цей пристрій можна побачити у форм-факторі звичайної банківської картки, але із певними відмінностями. В основному для документообігу та підтвердження використовують інші чіпи, що мають фнший функціонал. До прикладу, на них можна генерувати власні ключі для алгоритмів, щоб мінімізувати втручання роботу токену. Ось один із представників – Infenion sle-78clfx2400p.

2) Сматр-токен – це носій ключової інформації з USB інтерфейсом виконаний у металевому корпусі. Пристрій призначений для використання в системах інформаційного доступу, кваліфікованого електронного підпису, документообігу, інших системах для авторизації користувачів, захищеного зберігання та використання ключової інформації, а також може використовуватися в якості модуля безпеки в центрах сертифікації ключів та інших системах. Іншими словами, це є звичайною «флешкою», що використовує у собі, знову ж таки, спеціальний чіп, що вміє зберігати інформацію, редагувати її та виконувати криптографічні функції. Ось один із представників – Infenion sle-78cufx3000ph.

(а)

(б)

Рисунок 1.1 – Приклади готових токенів: (а) смарт-картки від Автор, (б) лінійка YubiKey-5 від Yubico.

2 ПОРІВНЯННЯ ІНТЕРФЕЙСІВ MICROSOFT CRYPTO API, PKCS11

Одразу почнімо із розбору основних інтерфейсів, що використовуються у криптографії. Визначимо спочатку для чого кожен інтерфейс використовується.

2.1 PKCS11

Стандарт інтерфейсу криптографічного маркера, PKCS11, створений компанією RSA Security і визначає власні інтерфейси програмування для криптографічних маркерів, таких як апаратні криптографічні прискорювачі та смарт-карти.

Почнімо із PKCS11. PKCS11 (або Cryptoki) є одним із стандартів криптографії з відкритим ключем, що відноситься до програмного інтерфейсу для створення та маніпулювання криптографічними токенами, де зберігається сектеринй ключ. Він був створений компанією RSA Security та у 1995 році було опубліковано першу весірю. Сам стандарт є незалежним від платформи API для криптографічних токенів, таких як апаратні модулі безпеки (HSM) і смарт-карти. Цей стандарт використовується у всіх можливих галузях від можливої автентифікації користувача на певній платформі до взаємоції із Центрами Сертифікації.

Основними функціями PKCS11 інтерфейсу моможна назвати:

– шифрування та розшифровування даних;

– генерація та перевірка цифрових підписів;

– ідентифікація і аутентифікація криптографічного модуля;

– управління ключами;

– виконання інших криптографічних функцій.

До прикладу використання цього інтерфейсу можна навести приклад

роботи із пристроєм від Yubico. Спробуємо вивести та показати сертифікат, що записаний на токен.

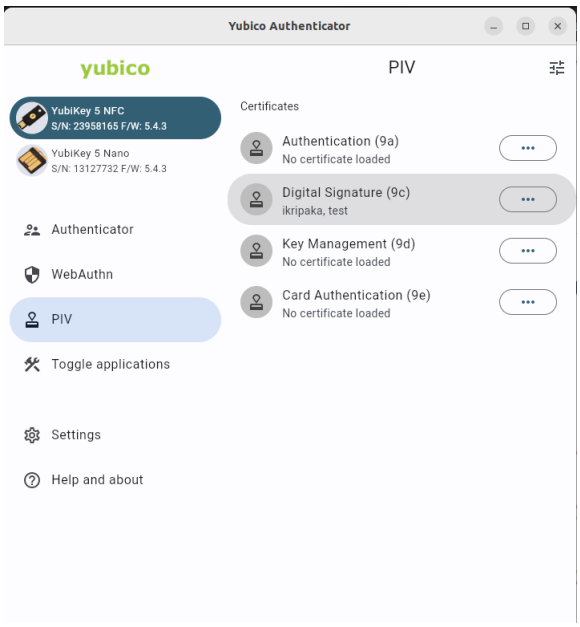

Рисунок 2.1 – Головний екран утиліти Yubico authenticator, де видно 2 токени та PIV(Personal Identity Verification).

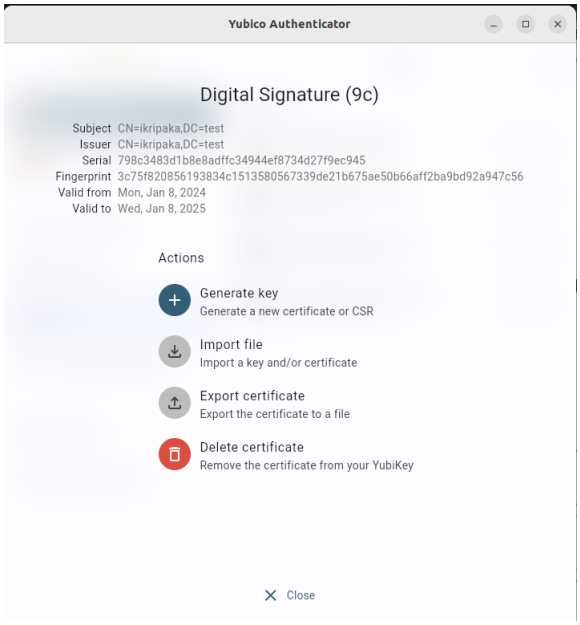

Рисунок 2.2 – Переглянемо ідентифікацію юзера за допомогою цифрового підпису.

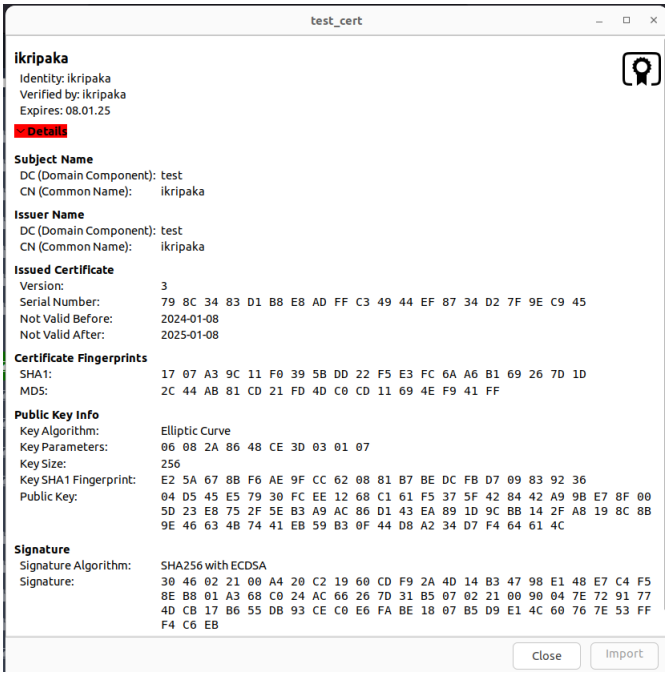

Рисунок 2.3 – Вигляд збереженого сертифікату через системний переглядач.

Коментуючи ці картинки можна сказати, що на першій картинці 2.1 видно головний екран Yubico Authenticator – програми, що дозволяє редагувати інформацію токена. Саме у цій версії можна побачити способи ідентифікаціїї. У даному випадку це ті ж WebAuth та PIV. Щодо другої 2.2, то там видно серійний номер, дату створення/час життя сртифікату. За допомогою можливості зберегти сертитфікат локально його вдалося переглянути і дізнатися як саме було згенеровано його 2.3.

2.2 Crypto API

Взагалі у сфері криптографії існує багато різних криптографічних інтерфейсів, але до заданого API Crypto API можна було знайти два можливих варіанти, один для операційної системи Windows, другий – для Linux. Розберемо їх та пояснимо різницю між ними. Почнемо із Microsoft CryptoAPI.

1) Microsoft CryptoAPI Microsoft CryptoAPI – це програмний

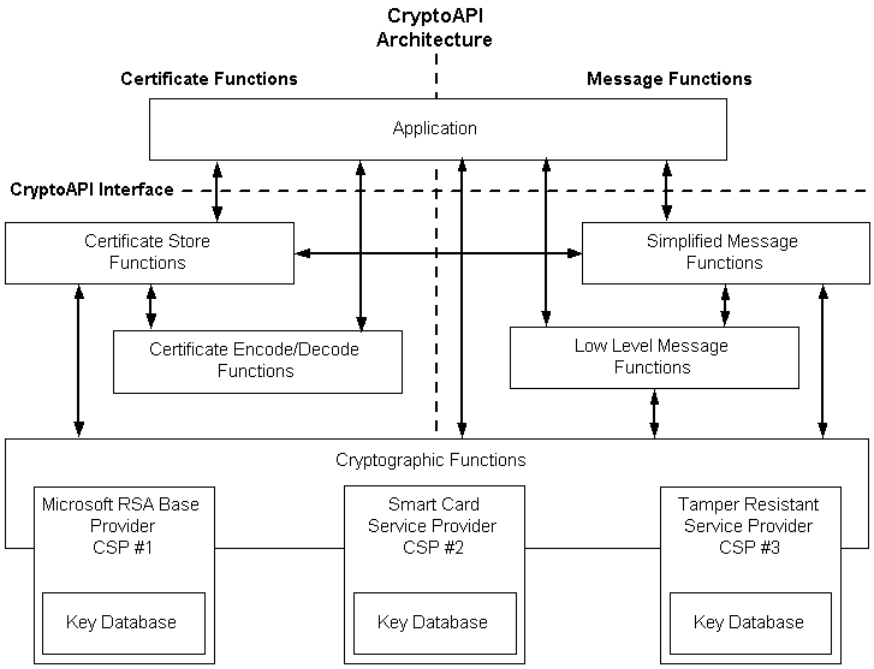

Рисунок 2.4 – Архітектура побудови CrypytoAPI.

інтерфейс прикладних програм (API), який надає розробникам Windows-додатків стандартний набір функцій для роботи з криптографією. Він входить до складу операційних систем Microsoft Windows і доступний для використання з простору користувача. Основною фішкою цього API є використання CSP (Cryptographic Service Providers), що є такими собі «незалежними» модулями. Тобто архітектура так побудована, що для взаємодії із будь-яким іншим пристроєм або модулем потрібно лише встановити бібліотеку для взаємодії із ним, а CryptoAPI сам зможе звернутися до пристрою і виконати відповідну операцію.

Основними функціями є Microsoft CryptoAPI:

– шифрування та розшифровування даних;

– генерація та перевірка цифрових підписів;

– алгоритми хеш-функції для перевірки цілісності даних;

– захист від сторонніх атак.

2) Crypto API (Linux) Crypto API - це криптографічна структура в ядрі Linux для різних частин ядра, які займаються криптографією, таких як IPsec і dm-crypt. Він був представлений у 2002 році і основною ціллю

було забезпечення внутрішніх потреб, однак крім самого ядра користь від нього може отримати і користувач. На даний час він має величезний функціонал із генерування симетричних/асиметричних ключів, взаємодії із різноманітними криптопримітивами. Детальніше можона знайти докумнтацію за покликанням.

Зауваження. – IPsec – це набір безпечних мережевих протоколів, який автентифікує та шифрує пакети даних, щоб забезпечити безпечний зашифрований зв'язок між двома комп'ютерами через мережу. Він використовується у віртуальних приватних мережах (VPN).

– dm-crypt – прозора підсистема шифрування блокових пристроїв у ядрі Linux версії 2.6 і пізніших. Воно є частиною інфраструктури пристрою відображення пристроїв і використовує криптографічні процедури з Crypto API ядра.

2.3 Відмінності CryptoAPI від PKCS11 інтерфейсів

Говорячи про відмінності, то можна їх класифікувати таким чином:

Характеристика	**Microsoft CryptoAPI**	**PKCS11**
Власність	власність Microsoft	відкритий стандарт
Доступність	простір користувача	простір ядра/простір користувача
Підтримувані алгоритми	широкий спектр	широкий спектрм
Гнучкість	менш гнучкий	більш гнучкий
Масштабованість	менш масштабований	більш масштабований
Використання	Windows	Windows, Linux, IOS, Android...

Коментуючи попередню таблицю можна сказати:

1) Щодо доступності, то CryptoAPI є історією так само локальною як і в попередньому пункті, але її звичайно можуть використовувати розробники/користувачі для певних потреб. Із іншого боку PKCS11 дає

лише інтерфейс, який усі повинні реалізувати, а «внутрощі» уже регламентуються виробниками та тим як вони записують дані.

2) Не можна не сказати слово про спосіб використання. Ці два порівнювані API можна сказзати взагалі із різних царин. Один використовується на портативних токенах для виконання криптографічниз операцій, а інший є архітектурою побудови/реалізації криптографічних програмних модулів. Такод додам, що сам інтерфейс CryptoAPI доступний лише на пристроях/програмному забезпеченні, що розробляє Microsoft, а, насамперед, PKCS11 стандарт доступний ледь не на кожній ОС. Останній є стандартом індустрії для реалізації HSM, смарт карток, токенів, що повинні використовуватися будь-де на будь-яких пристроях.

2.4 Основні задачі, що виникають при програмній реалізації криптосистем на токенах та сматр картках

На останок можна коротко розказати про основні проблеми, що можуть виникнути при реалізації криптосистем на токенах та сматри картках. Відповідно до проведеного дослідження проблеми можна класифікувати наступним чином.

– **Генератор псевдо випадкових чисел.** У токенах дуже важливо мати надійне джерело ентропії для генерування ключових пар.

– **Обмежена кількість пам'яті та обчислювальні можливості.** Так як токен чи смарт-картка мають досить малий форм-фактор, то до них не вдастся приєднати більшу кількість пам'яті. Тоді коли на комп'ютері можна сказаати «необмежені» можливості для обчислень, то у токенах розробникам приходиться максимально оптимізовувати їх реалізації.

ВИСНОВКИ

У звіті за темою «Дослідження основних задач що виникають при програмній реалізації криптосистем на інтелектуальних картках та токенах» було проведено порівняльний аналіз двох API: CryptoAPI та PKCS11. Роз'яснено та продемонстровоано приклад роботи із токенами від Yubikey. У результаті дізналися на практиці наскільки важливими вони є для автентифікації користувачів та провели маленький аналіз того, які виникають основні проблеми із реалізаціями крмптосистем на токенах.

ЛІТЕРАТУРА