МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»
Фізико-технічний інститут

ЛАБОРАТОРНА РОБОТА №4

з дисципліни
«МЕТОДИ РЕАЛІЗАЦІЇ КРИПТОГРАФІЧНИХ МЕХАНІЗМІВ»
на тему: «РОЗРОБКА РЕАЛІЗАЦІЇ АСИМЕТРИЧНОЇ КРИПТОСИСТЕМИ У ВІДПОВІДНОСТІ ДО СТАНДАРТНИХ ВИМОГ CRYPTO API АБО СТАНДАРТІВ PKCS»

Виконала:

студентка 6 курсу ФТІ
група ФБ-11мн
Чуракова Єкатєріна

Перевірила:
Байденко П.В.

Київ 2022

Варіант 2А: OPENSSL ПІД WINDOWS ПЛАТФОРМУ. КР/С ЕЛЬ ГАМАЛЯ

Web Crypto API — це рекомендація World Wide Web Consortium (W3C) щодо низькорівневого інтерфейсу, який підвищить безпеку веб-додатків, дозволяючи їм виконувати криптографічні функції без доступу до необробленого матеріалу ключа. Цей API виконуватиме базові криптографічні операції, такі як хешування, генерація підпису, перевірка та шифрування, а також дешифрування з веб-додатку.

Crypto API можна використовувати для широкого спектру цілей, зокрема:

- Забезпечення автентифікації для користувачів і служб
- Електронний підпис документів або коду
- Захист цілісності та конфіденційності зв'язку та обміну цифровими даними

Crypto API можна використовувати на будь-якій платформі. Це забезпечить загальний набір інтерфейсів, який дозволить веб-додаткам і прогресивним веб-додаткам виконувати криптографічні функції без необхідності доступу до необробленого матеріалу ключів. Додаткові інтерфейси в Crypto API дозволять генерувати ключі і виводити ключі.

Специфікація W3C для Crypto API зосереджується на загальній функціональності та функціях, які наразі існують між специфічними для платформи та стандартизованими криптографічними API порівняно з тими, які відомі лише кільком реалізаціям. Рекомендації групи щодо використання Crypto API не передбачають впровадження обов'язкового набору алгоритмів. Це пов'язано з усвідомленням того, що криптографічні реалізації відрізнятимуться серед відповідних агентів користувачів через державні постанови, місцеву політику, методи безпеки та проблеми інтелектуальної власності.

Існує багато типів існуючих веб-додатків, з якими Crypto API добре підходить для використання.

Багатофакторна аутентифікація

За допомогою Crypto API веб-програма матиме можливість забезпечувати автентифікацію зсередини себе замість того, щоб покладатися на автентифікацію на транспортному рівні для секретного ключа для автентифікації доступу користувача. Цей процес забезпечить багатший досвід для користувача.

Crypto API дозволить програмі знаходити відповідні ключі клієнта, які були раніше створені агентом користувача або були попередньо підготовлені веб-програмою. Програма зможе надати агенту користувача можливість генерувати новий ключ або повторно використовувати наявний ключ у випадку, якщо користувач не має ключа, уже пов'язаного з його обліковим записом.

Захищений обмін документами

API можна використовувати для захисту конфіденційних документів від несанкціонованого перегляду з веб-програми, навіть якщо вони були попередньо безпечно отримані. Веб-додаток використовуватиме Crypto API для шифрування

документа за допомогою секретного ключа, а потім оберне його відкритими ключами, пов'язаними з користувачами, які мають право переглядати документ. Після переходу до веб-додатку авторизований користувач отримає зашифрований документ і отримає вказівку використати свій закритий ключ, щоб розпочати процес розгортання, що дозволить йому розшифрувати та переглянути документ.

Хмарне сховище

Для захисту постачальник дистанційних послуг може захотіти, щоб їхні веб-програми надавали користувачам можливість захистити свої конфіденційні документи перед завантаженням своїх документів або інших даних. Crypto API дозволить користувачам:

- Вибрати особистий або секретний ключ
- Отримайте ключ шифрування з їх ключа
- Зашифрувати їхні документи/дані
- Завантажити їхні зашифровані документи/дані за допомогою існуючих API постачальника послуг

Підписання електронних документів

Багато веб-додатків приймають електронні підписи замість того, щоб вимагати письмові підписи. За допомогою Crypto API користувачеві буде запропоновано вибрати ключ, який можна буде згенерувати або попередньо підготувати спеціально для веб-програми. Потім ключ можна використовувати під час операції підписання.

Захист цілісності даних

Веб-програми часто кешують дані локально, що ставить дані під загрозу компрометації в разі офлайн-атаки. Crypto API дозволяє веб-додатку використовувати відкритий ключ, розгорнутий усередині нього, для перевірки цілісності кешу даних.

Безпечний обмін повідомленнями

API веб-криптографії може підвищити безпеку обміну повідомленнями для використання в автономних (OTR) та інших типах схем підписання повідомлень за допомогою угоди ключа. Відправник повідомлення та передбачуваний одержувач домовляться про спільні ключі шифрування та коду автентифікації повідомлення (MAC), щоб зашифрувати та розшифрувати повідомлення, щоб запобігти несанкціонованому доступу.

Підписування та шифрування об'єктів JSON (JSON)

Crypto API може використовуватися веб-додатками для взаємодії з форматами та структурами повідомлень, визначеними JSON Object Signing and Encryption (JOSE). Програма може зчитувати й імпортувати ключі веб-підпису JSON Web Signature (JWK), перевіряти повідомлення, захищені електронним підписом або MAC-ключами, і розшифровувати повідомлення JWE.

Отримані результати

```
Original message: Some message is present here!1
Encrypted message: a = 2B11BA0516165D0FB8898D156C52AECF0ED1E13E18CE27B107CD7ED88FFF4F2E55E0A3CA29EC8951DB51BA654D884E1EB
66B84E4EB049F0745A14580B652715459EF7D1E6B78E23FFFC0443D9A060717935276F5B490753E692F1503DE86C567B1D8775C81C9407E83CF1147A
2C43404E7A84253A9877CF0788498F6A1379C8E; b = A0C1C55AE9A2BC8BDB4BD9579EC120D881C7FEC113C59FD988D32FDADCD833A6802B56752FA
34B33DAE875812941D17B787E02EECA62F8B73E0CD516086A2DFC7A7629D0473E9660B2503F82AD20CEA33B9C7F2E37E41FFA879DF4D7A3EECEF4F39
94CF0E2552495AAF0062F38383C60A0D336EB145EE4FDF5072953510D653D
Decrypted message: Some message is present here!1
Signing message: Some message is present here!1
Signature: r = A6A6B93D6AC4B1E10C3F0B7DF3FCD0719AD6688A27553B9588B9DB5DAD5711C06AA92591A764EE09349FA68E8FC7148770C5F150F
7F2A7AFC18823EB3638450A2B507D9F1F1F8064DFB6CB13816F82D1035205D3850800896E7251606AAA236E80B4862851E7A6A8124C658F12F99821E
B8C83D4C174CEB6BA09B460925B8F53; s = 1DD0EB5CD2E6E864732881F0EC71508B104C1B56F7CC6D73F014745645FA043A21C2F482CB85D01283B
814356D28D2F26F8548A5D1ED2C48E0F4F38FC9BAB18F26F4338A1DED92ECB7DFAE7E3DE17E2ECBD66F00CE58E2C9934610B903903A6263936395A24
9E13893D6C43E16B474EEDAA4C2F27BA8B3327852397F6F6E98E8
Verify this message with signature: Some message is present here!1; Result: 1
Verify this message with signature: Some message is present here!2; Result: 0
```

Висновки

У роботі була реалізована обгортка на основі Crypto API для взаємодії із криптосистемою Ель-Гамаля, яка написана з використанням бібліотеки OpenSSL. Було протестовано роботу у режимі шифрування (шифрування і розшифрування), та у режимі підпису (підпис і верифікація).