**Проект e2ee\_project: Застосунок безпечного обміну даними**

**Основні бажані властивості безпеки застосунку**

* **Конфіденційність:** ніхто, окрім відправника та отримувача, не може отримати доступ до змісту повідомлення.
* **Цілісність:** будь-яке спотворення повідомлення буде виявлено.
* **Аутентифікація:** перевіряється справжність клієнта та сервера.
* **Швидкодія:** забезпечення високої продуктивність, особливо для систем із обмеженими ресурсами.
* **Динамічність:** зменшення ризику компрометації даних за рахунок ротації ключів.
* **Стійкість:** захист від атак, включно з MITM та атаками на протокол.

**Мотивація вибору криптографічної архітектури.**

**1. Аутентифікація серверів і клієнтів**

Для встановлення довіри між сторонами використовується **сертифікація X.509** в межах протоколу **TLS (SSL).**

**Чому TLS із сертифікатами?**

* + Автоматично забезпечує захист від MITM-атак.
  + Підтримує взаємну аутентифікацію (сертифікати як у клієнта, так і у сервера).
  + Використання перевірених стандартів (PKI) для шифрування та аутентифікації робить впровадження безпечним і зручним.

**2. Захист конфіденційності**

Для гарантії того, що лише відправник і отримувач можуть отримати доступ до повідомлень, використано **ChaCha20.**

**Чому ChaCha20?**

* + Швидший і енергоефективніший у порівнянні з AES, особливо для пристроїв із обмеженими ресурсами.
  + Висока криптографічна стійкість до атак, підтверджена численними дослідженнями.
  + Забезпечує захист конфіденційності навіть при динамічній зміні ключів.

**3. Цілісність повідомлень**

Для забезпечення цілісності повідомлень використовується **Poly1305.**

**Чому Poly1305?**

* + Оптимізований для спільного використання з ChaCha20.
  + Швидко генерує MAC (Message Authentication Code), який захищає дані від будь яких несанкціонованих змін.
  + Простий в інтеграції та має перевірену криптографічну безпеку.
  + Захищений від колізій та атак другого роду.

**4. Динамічна зміна ключів**

Для посилення безпеки використовується **періодична ротація ключів шифрування.**

**Як це працює?**

* + Ключі генеруються динамічно для кожної сесії.
  + Використовується **протокол узгодження ключів (ECDH)** на основі еліптичних кривих.
  + Зменшує ризик компрометації даних навіть у випадку компрометації одного ключа.

**Проект забезпечує стійкість до атак наступними інструментами**

**1. Захист від MITM-атак**

**TLS-з'єднання:**

* + Використання TLS забезпечує захищене з'єднання між клієнтом і сервером, унеможливлюючи прослуховування чи зміну трафіку третіми сторонами.
  + Всі клієнти та сервер використовують сертифікати X.509 для взаємної аутентифікації.
  + Підробити або перехопити сертифікат практично неможливо, якщо CA-сертифікат залишився конфіденційним.

**Перевірка сертифікатів:**

* + Клієнт і сервер перевіряють валідність сертифікатів за допомогою CA.
  + Якщо сертифікат недійсний або підроблений, з'єднання не встановлюється.

**2. Захист протоколу узгодження ключів**

**ECDH (Еліптичні криві):**

* + Протокол узгодження ключів ECDH гарантує, що навіть якщо зловмисник отримає доступ до зашифрованого трафіку, без відповідних приватних ключів він не зможе обчислити сесійний ключ.
  + Динамічна зміна ключів для ChaCha20 робить сесію короткочасною і захищає від повторних атак (replay attacks).

**3. Захист від атак на протокол**

**Використання ChaCha20 + Poly1305:**

* + Цей алгоритм забезпечує конфіденційність (ChaCha20) та автентичність (Poly1305).
  + Навіть якщо дані перехоплені і змінені, зловмисник не зможе створити правильну MAC для Poly1305, і повідомлення буде відхилено.

**Динамічна ротація ключів:**

* + Зменшує час дії сесійного ключа, що обмежує можливість атак на тривалі сесії (наприклад, brute-force).

**Виявлення модифікації даних:**

* + Клієнт і сервер перевіряють автентичність кожного повідомлення за допомогою MAC. Зміни в повідомленні викликають його відхилення.

**4. Захист від повторних атак**

**Унікальний ідентифікатор сесії:**

* + Кожна сесія отримує унікальний ідентифікатор, який виключає можливість повторного використання попереднього повідомлення в новій сесії.

**5. Жорстке логування**

* + Логи фіксують аномальні активності, що допомагає виявляти та аналізувати потенційні атаки.

**Не встиг**

1. Розширити протоколи безпеки для підтримки багатофакторної автентифікації.
2. Розробити інструменти для аналізу журналів і діагностики.