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

Звіт з виконання комп'ютерного практикуму

ДОСЛІДЖЕННЯ МЕТОДІВ РЕАЛІЗАЦІЇ СЛІПОГО ЦИФРОВОГО ПІДПИСУ

Виконали студенти
групи ФІ-32мн
Величко Олена,
Мельник Ілля,
Міснік Аліна

Перевірили:
Селюх П.В.

Київ — 2023

Мета роботи: Дослідити можливість реалізації криптографічного протоколу сліпого підпису. Порівняти ефективність роботи різних можливих реалізацій сліпого цифрового підпису

Постановка задачі: Розглянути та порівняти між собою алгоритми реалізації сліпого цифрового підпису: Rsa реалізація та за схемою Шнорра. Описати переваги та недоліки, можливі атаки на ці протоколи.

1 ХІД РОБОТИ

1.1 Що таке цифровий підпис

Цифровий підпис (ЦП) — це математичний метод, який використовується для перевірки автентичності та цілісності цифрового документа, повідомлення або програмного забезпечення. Це цифровий еквівалент власноручного підпису чи печатки, який забезпечує набагато більшу безпеку. Дійсний цифровий підпис на повідомленні дає одержувачу впевненість, що повідомлення надійшло від відправника відомого одержувачу.

Цифрові підписи є стандартним елементом більшості наборів криптографічних протоколів і зазвичай використовуються для розповсюдження програмного забезпечення, фінансових транзакцій, програмного забезпечення для керування контрактами та в інших випадках, коли важливо виявити підробку чи втручання.

Основні цілі і задачі ЦП:

1) Підтвердження цілісності повідомлення.

2) Підтвердження справжності джерела, автора повідомлення.

3) Забезпечення неможливості підробки ЦП.

4) Забезпечення неможливості відмови підписуючого від підписаного повідомлення.

5) Можливість багаторазової перевірки повідомлення без зміни криптосистеми різними користувачами в різний час.

6) Юридична значимість.

1.2 Сліпий цифровий підпис

Особливість сліпого цифрового підпису полягає у тому, що підписуюча сторона не знає зміст повідомлення, яке вона підписує. Поняття

сліпого підпису було створено Д. Шаумом у 1982 році, він й запропонував першу реалізацію цього підпису, яка базується на криптосистемі RSA.

Безпека схеми сліпого підпису ґрунтувалася на складності факторизації великих складених чисел.

Основна ідея сліпих підписів полягає в наступному:

1) Відправник **A** шифрує документ і надсилає його стороні **B**.

2) Сторона, не бачачи вміст документа, підписує його і повертає назад стороні **A**.

3) Сторона **A** знімає свій шифр, залишаючи на документі тільки підпис сторони **B**.

По завершенні цього протоколу сторона **B** нічого не знає ні про повідомлення t, ні про підписи під цим повідомленням.

Часто використовуваною аналогією з криптографічним сліпим підписом є фізична дія виборця, який вкладає заповнений анонімний бюлетень у спеціальний конверт, вистелений копіювальним папером, на зовнішній стороні якого попередньо надруковані облікові дані виборця. Посадова особа перевіряє повноваження та підписує конверт, таким чином переносячи свій підпис на бюлетень, що знаходиться всередині, через копіювальний папір. Після підписання пакет повертається виборцю, який перекладає підписаний бюлетень у новий звичайний конверт без позначок. Таким чином, підписувач не переглядає вміст повідомлення, але третя сторона може пізніше перевірити підпис і знати, що підпис дійсний в межах обмежень основної схеми підпису.

Мета сліпого підпису полягає в тому, щоб перешкодити підписувачу **B** ознайомитися з повідомленням сторони **A**, яку він підписує, і з відповідним підписом під цим повідомленням. Тому надалі підписане повідомлення неможливо пов'язати зі стороною **A**.

Схема безпечного сліпого підпису повинна задовольняти 3 властивостям, а саме:

1) **Нульове розголошування**. Ця властивість допомагає користувачеві отримати підпис на даному повідомленні, не розкриваючи

самого повідомлення підписуючій стороні.

2) **Невідстежуваність**. Підписуюча сторона не може відстежити пару підпис-повідомлення після того, як користувач оприлюднив підпис на повідомленні.

3) **Непідкладність**. Тільки підписуча сторона може сгенерувати дійсний підпис. Ця властивість найважливіша і повинна задовольнятися для всіх схем підписів.

Завдяки властивостям нульового розголошення і невідстежуваності, схема сліпого підпису може бути широко задіяна в додатках, де необхідна конфіденційність, наприклад, в системах електронного голосування.

1.2.1 Сліпий цифровий підпис на основі RSA

Нехай **A** має RSA з відкритими параметрами (n, e) та секретом d та нехай M, $M < n$ – повідомлення від користувача **B**.

Задача сліпого цифрового підпису: **A** повинен підписати M, не знаючи про нього ніякої інформації (тобто щоб **A** не зміг прочитати повідомлення).

Алгоритм формування сліпого цифрового підпису на основі RSA буде виглядати таким чином:

1) **B**: обирає випадкове r таке, що $1 < r < n-1$, $(r, n) = 1$, та обчислює $r^e \mod n = k$ – «засліплюючий» множник та $\widetilde{M} = M \cdot k \mod n$, і відправляє $\widetilde{M}$ користувачу **A** для підписання.

2) **A** підписує засліплене повідомлення: $\widetilde{M^d} \mod n = \widetilde{S}$ та відправляє пару $(\widetilde{M}, \widetilde{S})$ користувачу **B**.

3) **B** обчислює $\widetilde{S} = \widetilde{M^d} \mod n = M^d \cdot k^d \mod n = M^d \times$
$\times r^{e \cdot d} \mod n = M^d \cdot r \mod n$ та знімає засліплення: $M^d = \widetilde{S} \cdot r^{-1} \mod n = S$. Таким чином формується повідомлення (M, S) зі сліпим цифровим підписом.

1.2.2 Сліпий підпис на основі ЕЦП Шнорра

Нехай Аліса хоче підписати повідомлення m у Боба таким чином, щоб, по-перше, Боб не міг ознайомитися з повідомленням в ході підпису, по-друге, щоб Боб не міг згодом при отриманні повідомлення m і відповідного підпису ідентифікувати користувача, який ініціював протокол сліпого підпису для даного конкретного повідомлення. Протокол реалізується наступним чином:

1) Аліса ініціює взаємодію з Бобом.

2) Боб відправляє Алісі значення $R = a^k \mod p$.

3) Аліса обчислює значення $R' = Ra^{-w}y^{-t} \mod y$ (w і t — випадкові числа, що не перевищують y), $E' = H(m||R')$ і $E = E' + t \mod y$, після чого відправляє Бобу значення E.

4) Боб обчислює значення S, таке що $R = a^S y^E \mod p$, і відправляє S Алісі.

5) Аліса обчислює підпис (E', S'), де $E' = E^{-t} \mod y$ і $S' = S - - w \mod y$, який є справжнім по відношенню до повідомлення m.

1.2.3 Уразливості сліпого підпису

Алгоритм RSA може бути об'єктом атаки, завдяки якій стає можливим розшифрувати раніше підписане наосліп повідомлення, видавши його за повідомлення, яке тільки ще треба підписати. Виходячи з того, що процес підпису еквівалентний розшифровці підписуючою стороною (з використанням секретного ключа), атакуючий може підкласти для підпису вже підписану наосліп версію повідомлення m, зашифрованого за допомогою відкритого ключа підписуючої сторони, тобто підкласти повідомлення m'.

$$\begin{aligned} m'' &= m'r^e \pmod n \\ &= (m^e \pmod n \cdot r^e) \pmod n \\ &= (mr)^e \pmod n \end{aligned}$$

де m' — це зашифрована версія повідомлення. Коли повідомлення підписане, відкритий текст m легко отримуємо:

$$\begin{aligned} s' &= m''^d \pmod n \\ &= ((mr)^e \pmod n)^d \pmod n \\ &= (mr)^{ed} \pmod n \\ &= m \cdot r \pmod n, \text{ since } ed \equiv 1 \pmod{\phi(n)} \end{aligned}$$

де $\phi(n)$ — це Функція Ейлера. Тепер повідомлення легко отримати.

$$m = s' \cdot r^{-1} \pmod n$$

Атака працює, тому що в цій схемі підписуюча сторона підписує безпосередньо саме повідомлення, а у звичайних схемах підпису підписуюча сторона зазвичай підписує, наприклад, криптографічну хеш-функцію. Тому через цю мультиплікативну властивість RSA, один ключ ніколи не повинен використовуватися одночасно для шифрування і підписання наосліп.

1.3 Порівняння алгоритмів сліпого цифрового підпису на основі RSA і Шнорра. Проблематика реалізації

Нам вдалося реалізувати алгоритм сліпого підпису на основі RSA. Ця реалізація є доволі простою, якщо мати бібліотеку з підтримкою реалізації великої арифметики та швидкого піднесення до великого степеня числа за модулем. До прикладу наша реалізація, з використанням бібліотеки $BigInteger$, відпрацьовує за $6,7 \cdot 10^{-9}$ с, причому на формуванні маски ми затрачаємо всього $0,3 \cdot 10^{-9}$ с. Дані оцінки були отримані для чисел

довжини 1024 біт. І хоча даний алгоритм є доволі простий в реалізації та ефективний для реалізації, особливо в малих обчислювальних ресурсах, ми повинні згадати про атаку, що була наведена вище.

Алгоритм Шнорра виявився не дуже простим для реалізації, оскільки при розгортанні цього алгоритму, а саме під час генерування ключів – потрібно згенерувати такі прості числа p та q, що $q|p-1$, а також $a \in \mathbb{Z}_p$ таке, що $a^q = 1 \mod p$ і $a \neq 1$. Ми вирішили генерувати спочатку q за допомогою Блюм-Блюма-Шуба, перевіряти на простоту, і якщо число просте – генерувати другий множник так, щоб отримати $p-1$. Найбільшою проблемою виявилося отримати генератор або елемент з порядком q, оскільки його обчислювально важко отримати в групі з великими порядком. Очевидно, що якщо $p \geqslant 2^{512}$, то немає сенсу перебирати всі елементи групи, а знайти корінь степеня q також є проблематичним.

ВИСНОВКИ

У даній роботі ми дослідили можливість реалізації таких алгоритмів сліпого підпису як RSA та Шнорра. Реалізація алгоритму RSA може бути досить успішною за наявності бібліотек багаторозрядної арифметики та є досить простою для побудови. Основним недоліком цього алгоритму є можливість атаки з використанням шифротексту замість повідомлення. Алгоритм Шнорра, навпаки, стійкий до цієї атаки, оскільки використовує гешування повідомлення. Також для реалізації він має серйозний недолік, який полягає у пошуку елемента великого порядку під час генерації ключів.