Практическое занятие №7

ЭЛЕКТРОННО-ЦИФРОВАЯ ПОДПИСЬ

Дисциплина: Основы информационной безопасности

Выполнил: Студент 2 курса 1 группы Васильев В. В.

Проверил: Ст. преп. Ржеуцкая Н. В.

**Цель занятия**  
Целью данного занятия является получение практических знаний и навыков по использованию методов криптографии для создания и проверки электронной цифровой подписи (ЭЦП). В ходе выполнения задания будет рассмотрен алгоритм RSA, который является основой многих современных криптографических систем. Особое внимание будет уделено теоретическим аспектам и пошаговому объяснению процедур, а также реальной реализации ЭЦП с использованием данного алгоритма.

**Краткие теоретические сведения**

Электронная цифровая подпись (ЭЦП) — это криптографический механизм, который используется для проверки подлинности и целостности данных. ЭЦП основывается на принципах асимметричного шифрования, где используется пара ключей: один — для подписания, второй — для проверки подписи. Асимметричные алгоритмы позволяют обеспечить высокую степень безопасности, поскольку для шифрования данных используется один ключ, а для их расшифровки — другой.

Алгоритм RSA (Ривест, Шамир, Адлеман) является одним из самых популярных асимметричных алгоритмов шифрования. Он основан на сложности факторизации больших чисел. Основной принцип работы алгоритма заключается в генерации пары ключей, которая используется для шифрования и расшифрования данных или для создания и проверки подписи.

Процесс подписания с помощью RSA включает два этапа:

1. Генерация хэш-суммы сообщения.
2. Подписание хэш-суммы с использованием закрытого ключа.

Для проверки подписи отправитель должен использовать свой открытый ключ. Если хэш-сумма, полученная при расшифровке подписи, совпадает с хэш-суммой сообщения, то подпись считается верной.

**Условие задания**

1. Объяснить последовательность выполнения процедур генерации и проверки ЭЦП.
2. Описать последовательность действий участников протокола при отправке и проверке ЭЦП.
3. Описать схему протокола ЭЦП на основе алгоритма RSA.

**Исполнительная часть**

1. **Процесс генерации и проверки ЭЦП**

Процесс создания и проверки ЭЦП состоит из нескольких ключевых этапов:

1. Генерация ключей:
   * На первом этапе генерируются два ключа: закрытый и открытый. Закрытый ключ используется для подписания, а открытый — для проверки подписи. Ключи генерируются с использованием алгоритма RSA, который включает в себя выбор двух больших простых чисел, вычисление их произведения и применение других математических операций.
   * Генерация происходит следующим образом:
     + Выбираются два больших простых числа ***p*** и ***q***.
     + Вычисляется их произведение ***n = p × q***, которое используется в качестве модуля для открытого и закрытого ключей.
     + Вычисляется функция Эйлера ***ϕ(n)=(p−1)(q−1)***
     + Далее выбирается число ***e***, которое должно быть взаимно простым с ***ϕ(n).***
     + Вычисляется ***d***, которое является мультипликативной инверсией ***e*** по модулю ***ϕ(n).***
   * Таким образом, открытый ключ состоит из пары (e, n), а закрытый — из пары (d, n).
2. Генерация подписи:
   * Для того чтобы подписать сообщение, сначала нужно вычислить его хэш-сумму, используя криптографическую хеш-функцию (например, SHA-256). Хэш-сумма — это фиксированная длина строки, которая уникально представляет собой содержимое сообщения.
   * После того как хэш-сумма сообщения вычислена, она подписывается закрытым ключом отправителя. Подпись — это результат шифрования хэш-суммы с использованием закрытого ключа.
   * Подписанное сообщение отправляется получателю вместе с подписью.
3. Проверка подписи:
   * Получатель получает сообщение и подпись. Чтобы проверить подпись, он использует открытый ключ отправителя.
   * Сначала получатель вычисляет хэш-сумму из полученного сообщения.
   * Затем он расшифровывает подпись с помощью открытого ключа отправителя. Полученная при этом хэш-сумма должна совпасть с хэш-суммой сообщения.
   * Если хэш-суммы совпадают, это подтверждает подлинность подписи.
4. **Действия участников протокола**
5. **Отправитель**:
   * Отправитель генерирует пару ключей с использованием алгоритма RSA.
   * Создает хэш-сумму сообщения с помощью криптографической хеш-функции.
   * Подписывает хэш-сумму закрытым ключом.
   * Отправляет сообщение и подпись получателю.
6. **Получатель**:
   * Получает сообщение и подпись.
   * Вычисляет хэш-сумму из полученного сообщения.
   * Расшифровывает подпись с помощью открытого ключа отправителя.
   * Сравнивает полученную хэш-сумму с вычисленной. Если они совпадают, то сообщение и подпись являются подлинными.
7. **Схема протокола ЭЦП на основе алгоритма RSA**

Протокол ЭЦП может быть представлен следующими этапами:

1. **Генерация ключей**:
   * Отправитель генерирует пару ключей: (e, n) — открытый, (d, n) — закрытый.
2. **Создание подписи**:
   * Отправитель вычисляет хэш-сумму сообщения.
   * Отправитель подписывает хэш-сумму с использованием закрытого ключа.
3. **Отправка сообщения**:
   * Отправитель отправляет сообщение и подпись получателю.
4. **Проверка подписи**:
   * Получатель вычисляет хэш-сумму сообщения.
   * Получатель расшифровывает подпись с использованием открытого ключа отправителя.
   * Если хэш-сумма совпадает, сообщение считается подлинным.

**Использованные источники**

1. Криптографический стандарт RSA, <https://www.rsa.com>
2. Учебник "Криптография и безопасность информации",
3. Официальная документация библиотеки PyCryptodome, <https://www.pycryptodome.org>
4. Статья "Как работает RSA и его применение в криптографии".