Алгоритм электронной цифровой подписи на основе решения системы сравнений

СОДЕРЖАНИЕ:

ВВЕДЕНИЕ

В последнее время все больше и больше внедряются в нашу повседневную жизнь информационные технологии, пытаясь захватить в ней все: от важнейших государственных проектов до решения обычных бытовых проблем. Вместе с огромной пользой и, казалось бы, неограниченными возможностями новые технологии приносят и новые проблемы. Одной из них является проблема защиты информации от несанкционированного посягательства теми, кто доступа к этой информации иметь не должен. В связи с этим почти одновременно с развитием информационных и компьютерных технологий начали развиваться и технологии защиты информации, развитие которых с некоторой точки зрения гораздо более критично, чем развитие непосредственно информационных технологий. Ведь с совершенствованием систем защиты, совершенствуются и методы взлома, обхода этих защит, что требует постоянного пересмотра и увеличения надежности защиты информации.

Актуальность темы. Электронная цифровая подпись – это эффективное средство защиты информации от модификации, искажений, позволяющее при этом однозначно идентифицировать отправителя сообщения и перенести свойства реальной подписи под документом в область электронного документа. Электронная цифровая подпись является наиболее перспективным и широко используемым в мире способом защиты электронных документов от подделки и обеспечивает высокую достоверность сообщения.

Данная курсовая работа посвящена одной из важнейших задач криптографии – электронной цифровой подписи (digital signature). Электронная цифровая подпись (ЭЦП) необходима для однозначного и никем неоспоримого установления автора какого-либо документа. Фактически, ЭЦП служит аналогом обычной подписи, которая устанавливает подлинность какого-либо документа или договора. Но поскольку в последнее

время огромное количество договоров и документов заключаются с использованием электронных и компьютерных средств, то поставить на них обычную подпись не представляется возможным. Именно для таких ситуаций и используется электронная цифровая подпись. Электронная цифровая подпись создана для того, чтобы избежать подделок, а также искажений передаваемых сообщений.

Целью данной курсовой работы является рассмотреть алгоритм электронной цифровой подписи на основе решения системы сравнений. Для достижения поставленной цели необходимо решить ряд **задач:**

- раскрыть основные понятия связанные с ЭЦП;
- показать назначение и применение ЭЦП;
- провести обзор алгоритмов ЭЦП;
- рассмотреть схемы ЭЦП с новым механизмом формирования подписи, а именно: схема цифровой подписи ElGamal; схема цифровой подписи RSA; схема цифровой подписи ECDSA;
- исследовать схемы с формированием подписи на основе решения системы сравнений.

Курсовая работа состоит из введения, двух глав, включающих в себя шесть параграфов, заключения и списка литературы.

ГЛАВЫ 1. ЭЛЕКТРОННАЯ ЦИФРОВАЯ ПОДПИСЬ: ОСНОВНЫЕ ПОНЯТИЯ И ОПРЕДЕЛЕНИЯ

1.1. Основные понятия связанные с ЭЦП. Назначение и применение ЭЦП

(Федеральный закон Российской Федерации от 6 апреля 2011 г. N 63-ФЗ "Об электронной подписи")[1].

1) **электронная подпись** - информация в электронной форме, которая присоединена к другой информации в электронной форме (подписываемой информации) или иным образом связана с такой информацией и которая используется для определения лица, подписывающего информацию;

2) **сертификат ключа проверки электронной подписи** - электронный документ или документ на бумажном носителе, выданные удостоверяющим центром либо доверенным лицом удостоверяющего центра и подтверждающие принадлежность ключа проверки электронной подписи владельцу сертификата ключа проверки электронной подписи;

3) **квалифицированный сертификат ключа проверки электронной подписи** (далее - квалифицированный сертификат) - сертификат ключа проверки электронной подписи, выданный аккредитованным удостоверяющим центром или доверенным лицом аккредитованного удостоверяющего центра либо федеральным органом исполнительной власти, уполномоченным в сфере использования электронной подписи (далее - уполномоченный федеральный орган);

4) **владелец сертификата ключа проверки электронной подписи** - лицо, которому в установленном настоящим Федеральным законом порядке выдан сертификат ключа проверки электронной подписи;

5) **ключ электронной подписи** - уникальная последовательность символов, предназначенная для создания электронной подписи;

[1] Федеральный закон от 06.04.2011 N 63-ФЗ (ред. от 19.12.2022) «Об электронной подписи» – URL: https://www.consultant.ru/document/cons_doc_LAW_112701/ (дата обращения: 20.12.2022).

6) **ключ проверки электронной подписи** - уникальная последовательность символов, однозначно связанная с ключом электронной подписи и предназначенная для проверки подлинности электронной подписи;

7) **удостоверяющий центр** - юридическое лицо или индивидуальный предприниматель, осуществляющие функции по созданию и выдаче сертификатов ключей проверки электронных подписей, а также иные функции, предусмотренные настоящим Федеральным законом;

8) **аккредитация удостоверяющего центра** - признание уполномоченным федеральным органом соответствия удостоверяющего центра требованиям настоящего Федерального закона;

9) **средства электронной подписи** - шифровальные (криптографические) средства, используемые для реализации хотя бы одной из следующих функций - создание электронной подписи, проверка электронной подписи, создание ключа электронной подписи и ключа проверки электронной подписи;

10) **средства удостоверяющего центра** - программные и (или) аппаратные средства, используемые для реализации функций удостоверяющего центра;

11) **участники электронного взаимодействия** - осуществляющие обмен информацией в электронной форме государственные органы, органы местного самоуправления, организации, а также граждане;

12) **корпоративная информационная система** - информационная система, участники электронного взаимодействия в которой составляют определенный круг лиц;

13) **информационная система общего пользования** - информационная система, участники электронного взаимодействия в которой составляют неопределенный круг лиц и в использовании которой этим лицам не может быть отказано.

Цифровая подпись предназначена для аутентификации лица, подписавшего электронный документ. Кроме этого, использование цифровой подписи позволяет осуществить:

Контроль целостности передаваемого документа: при любом случайном или преднамеренном изменении документа подпись станет недействительной, потому что вычислена она на основании исходного состояния документа и соответствует лишь ему.

Защиту от изменений (подделки) документа: гарантия выявления подделки при контроле целостности делает подделывание нецелесообразным в большинстве случаев.

Невозможность отказа от авторства. Так как создать корректную подпись можно, лишь зная закрытый ключ, а он должен быть известен только владельцу, то владелец не может отказаться от своей подписи под документом.

Доказательное подтверждение авторства документа: Так как создать корректную подпись можно, лишь зная закрытый ключ, а он должен быть известен только владельцу, то владелец пары ключей может доказать своё авторство подписи под документом. В зависимости от деталей определения документа могут быть подписаны такие поля, как «автор», «внесённые изменения», «метка времени» и т.д.[2]

Использование хэш-функций

Поскольку подписываемые документы — переменного (и как правило достаточно большого) объёма, в схемах ЭЦП зачастую подпись ставится не на сам документ, а на его хэш. Для вычисления хэша используются криптографические хэш-функции, что гарантирует выявление изменений документа при проверке подписи. Хэш-функции не являются частью алгоритма ЭЦП, поэтому в схеме может быть использована любая надёжная хэш-функция.

[2] ГОСТ Р 34.10-2001. Информационные технологии. Криптографическая защита информации. Процессы формирования и проверки цифровой подписи.

Использование хэш-функций даёт следующие преимущества:

1. **Вычислительная сложность**. Обычно хеш цифрового документа делается во много раз меньшего объёма, чем объём исходного документа, и алгоритмы вычисления хеша являются более быстрыми, чем алгоритмы ЭЦП. Поэтому формировать хэш документ и подписывать его получается намного быстрее, чем подписывать сам документ.

2. **Совместимость**. Большинство алгоритмов оперирует со строками бит данных, но некоторые используют другие представления. Хеш-функцию можно использовать для преобразования произвольного входного текста в подходящий формат.

3. **Целостность**. Без использования хеш-функции большой электронный документ в некоторых схемах нужно разделять на достаточно малые блоки для применения ЭЦП. При верификации невозможно определить, все ли блоки получены и в правильном ли они порядке.

Стоит заметить, что использование хеш-функции не обязательно при цифровой подписи, а сама функция не является частью алгоритма ЭЦП, поэтому хеш-функция может использоваться любая или не использоваться вообще.

1.2. Обзор алгоритмов ЭЦП

Технология применения системы электронной цифровой подписи (ЭЦП) предполагает наличие сети абонентов, посылающих друг другу подписанные электронные документы. Для каждого абонента генерируется пара ключей: секретный и открытый. Секретный ключ хранится абонентом в тайне и используется им для формирования ЭЦП. Открытый ключ известен всем другим пользователям и предназначен для проверки ЭЦП получателем подписанного электронного документа. Иначе говоря, открытый ключ является необходимым инструментом, позволяющим проверить подлинность электронного документа и автора подписи. Открытый ключ не позволяет вычислить секретный ключ.

Для генерации пары ключей (секретного и открытого) в алгоритмах ЭЦП, используются разные математические схемы, основанные на применении однонаправленных функций. Эти схемы разделяются на две группы. В основе такого разделения лежат известные сложные вычислительные задачи:

- задача факторизации (разложения на множители) больших целых чисел;
- задача дискретного логарифмирования[3].

Первой и наиболее известной во всем мире конкретной системой ЭЦП стала система **RSA**, математическая схема которой была разработана в 1977 г. В Массачуссетском технологическом институте США. Алгоритм получил свое название по первым буквам фамилий его авторов: Rivest, Shamir и Adleman. Надежность алгоритма основывается на трудности факторизации больших чисел.

Более надежный и удобный для реализации на персональных компьютерах ЭЦП алгоритм был разработан в 1984 г. американцем арабского происхождения Тахером Эль Гамалем и получил название **El Gamal Signature Algorithm (EGSA)**.

Идея EGSA основана на том, что для обоснования практической невозможности фальсификации ЭЦП может быть использована более сложная вычислительная задача, чем разложение на множители большого целого числа – задача дискретного логарифмирования. Кроме того Эль Гамалю удалось избежать явной слабости алгоритма ЭЦП RSA, связанной с возможностью подделки ЭЦП под некоторыми сообщениями без определения секретного ключа.

Алгоритм цифровой подписи **Digital Signature Algorithm (DSA)** предложен в 1991г. в США для использования в стандарте цифровой подписи DSS (Digital Signature Standard). Алгоритм DSA является развитием алгоритма ЭЦП EGSA. По сравнению с алгоритмом ЭЦП EGSA алгоритм

[3] ГОСТ Р 34.10-94. Информационные технологии. Криптографическая защита информации. Процедуры выработки и проверки электронной цифровой подписи на базе асимметричного криптографического алгоритма

DSA имеет ряд преимуществ: сокращен объем памяти и время вычисления подписи. Недостатком же является необходимость при подписывании и проверке подписи выполнять сложные операции деления по модулю большого числа.

Российский стандарт цифровой подписи обозначается как **ГОСТ Р 34.10-94**. Алгоритм цифровой подписи, определяемый этим стандартом, концептуально близок к алгоритму DSA. Различие между этими стандартами заключается в использовании параметров ЭЦП разного порядка, что приводит к получению более безопасной подписи при использовании российского стандарта.

Алгоритмы цифровых подписей **Elliptic Curve Digital Signature Algorithm (ECDSA)** и **ГОСТ Р 34.10-2001** являются усовершенствованием цифровых подписей DSA и ГОСТ Р 34.10-94 соответственно. Эти алгоритмы построены на базе математического аппарата эллиптических кривых над простым полем Галуа[4].

[4] Игоничкина Е.В. Анализ алгоритмов электронной цифровой подписи – URL: https://security.ase.md/publ/ru/pubru86/ (дата обращения: 20.12.2022).

ГЛАВА 2. СХЕМЫ ЭЦП С НОВЫМ МЕХАНИЗМОМ ФОРМИРОВАНИЯ ПОДПИСИ

2.1. Схема цифровой подписи ElGamal

История: Схема была предложена Тахером Эль - Гамалем в 1984 году.

Эль - Гамаль разработал один из вариантов алгоритма Диффи - Хеллмана. Он усовершенствовал систему Диффи - Хеллмана и получил два алгоритма, которые использовались для шифрования и для обеспечения аутентификации. В отличии от RSA алгоритм Эль-Гамаля не был запатентован и, поэтому, стал более дешевой альтернативой, так как не требовалась оплата взносов за лицензию.

Схема Эль - Гамаля - криптосистема с открытым ключом, основанная на трудности вычисления дискретных логарифмов в конечном поле. Криптосистема включает в себя алгоритм шифрования и алгоритм цифровой подписи. Схема Эль - Гамаля лежит в основе стандартов электронной цифровой подписи в США и России[5].

Генерация ключей: Процедура генерации ключей здесь точно такая же, как та, которая используется в криптографической системе.

1. Генерируется случайное простое число **P** длины n битов.
2. Выбирается произвольное целое число **g**, являющееся первообразным корнем по модулю **P**.
3. Выбирается случайное целое число **x** такое, что $\mathbf{1 < x < P}$.
4. Вычисляется $\mathbf{y = g^x \bmod p}$.
5. Открытым ключом является тройка (**p**, **g**, **y**), закрытым ключом – **x.**

Подписание:

1. Вычисляется хэш сообщения **M**: **m = h(M)**.

[5] Семёнов, Г. Цифровая подпись. Эллиптические кривые. / Г. Семёнов. // Открытые системы. – № 7-8/2022. – С. 98.

2. Выбирается случайное число **1 < k < p – 1** взаимно простое с **p - 1** и вычисляется $\mathbf{r = g^k \bmod p}$.

3. С помощью расширенного алгоритма Евклида вычисляется число **s**, удовлетворяющее сравнению: **m ≡ xr + ks (mod p – 1)**.

4. Подписью сообщения **M** является пара (**r**, **s**).

Проверка: Зная открытый ключ (**p**, **g**, **y**), подпись (**r**, **s**) сообщения **M** проверяется следующим образом:

1. Проверяется выполнимость условий: **0 < r < p** и **0 < s < p - 1**. Если хотя бы одно из них не выполняется,то подпись считается неверной.

2. Вычисляется хэш **m = h(M)**.

3. Подпись считается верной, если выполняется сравнение: $\mathbf{g^r\ r^s \bmod p \equiv g^m \bmod p}$.

5. Схема цифровой подписи Шнорра

Проблема схемы цифровой подписи Эль - Гамаля в том, что **p** должно быть очень большим, чтобы сделать трудной проблему дискретного логарифма **Zp***. Рекомендуется длина **p** по крайней мере 1024 битов. Можно сделать подпись размером 2048 бит. Чтобы уменьшить размер подписи, Шнорр предложил новую схему, основанную на схеме Эль-Гамаля , но с уменьшенным размером подписи.

Генерация ключей: Перед подписанием сообщения Алиса должна генерировать ключи и объявить всем общедоступные ключи.

1. Алиса выбирает простое число **p**, которое обычно равно по длине 1024 битам.

2. Алиса выбирает другое простое число **q**, которое имеет тот же самый размер, что и хэш, созданный функцией криптографического хэширования (≥160). **q | (p - 1)**, другими словами, **(p - 1) = 0 mod q**.

3. Алиса выбирает $\mathbf{e_1}$, **q**-тый корень которого был бы равен **1 mod p**. Чтобы сделать это, Алиса выбирает примитивный элемент в **Zp**, $\mathbf{e_0}$ и вычисляет $\mathbf{e_1 = e_0^{(p-1)/q} \bmod p}$.

4. Алиса выбирает целое число **d**, как свой секретный ключ.

5. Алиса вычисляет $\mathbf{e_2 = e_1^d \bmod p}$.

Общедоступный ключ Алисы - ($\mathbf{e_1}$, $\mathbf{e_2}$, **p**, **q**), ее секретный ключ - (**d**).

Подписание:

1. Алиса выбирает случайное число **r**. (меняется каждый раз при новом сообщении) такое что $\mathbf{1 < r < q}$.

2. Алиса вычисляет первую подпись $\mathbf{S_1 = h(M \mid e_1^r \bmod p)}$. Сообщение присоединяется спереди к значению $\mathbf{e_1^r \bmod p}$, затем применяется хэш-функция, чтобы создать хэш. хэш-функция непосредственно не применяется к сообщению, но вместо этого она получается из последовательного соединения **M** и $\mathbf{e_1^r \bmod p}$.

3. Алиса вычисляет вторую подпись $\mathbf{S_2 = r + d * S_1 \bmod q}$.

4. Алиса передает **M**, $\mathbf{S_1}$ и $\mathbf{S_2}$.

Проверка: Приемник, например Боб, получает **M**, $\mathbf{S_1}$ и $\mathbf{S_2}$.

1. Боб вычисляет $\mathbf{V = h\,(M \mid e_1^{S2} * e_2^{-S1} \bmod p)}$.

2. Если $\mathbf{S_2}$ сравнимо с **V** по модулю **p**, сообщение принято; иначе оно отклоняется.

2.2. Схема цифровой подписи RSA

История: Описание RSA было опубликовано в 1977 году Рональдом Райвестом (Ronald Linn Rivest), Ади Шамиром (Adi Shamir) и Леонардом Адлеманом (Leonard Adleman) из Массачусетского Технологического Института (MIT).Британский математик Клиффорд Кокс (Clifford Cocks), работавший в центре правительственной связи (GCHQ) Великобритании, описал аналогичную систему в 1973 году во внутренних документах центра, но эта работа не была раскрыта до 1977 года и Райвест, Шамир и Адлеман разработали RSA независимо от работы Кокса[6].

Описание алгоритма: Безопасность алгоритма электронной подписи RSA основана на трудности задачи разложения на множители. Алгоритм

[6] Полянская, О.Ю. Инфраструктура открытых ключей. Учебное пособие. / О.Ю. Полянская, В.С. Горбатов. – Москва, 2017. – С. 123.

использует два ключа — открытый и секретный, вместе открытый и соответствующий ему секретный ключи образуют пару ключей. Открытый ключ не требуется сохранять в тайне, он используется для зашифровывания данных. Если сообщение было зашифровано открытым ключом, то расшифровать его можно только соответствующим секретным ключом. RSA может также применяться для того, чтобы подписать и подтвердить сообщение. Схема цифровой подписи меняет роли секретных и открытых ключей. Первое: используются секретный и открытый ключи передатчика, а не приемника. Второе: передатчик использует свой собственный секретный ключ для подписи документа; приемник использует открытый ключ передатчика, чтобы проверить этот документ.

Генерация ключей: Генерация ключей в схеме цифровой подписи RSA точно такая же, как и генерация ключей в криптографической системе RSA. Алиса выбирает два простых числа **p** и **q** и вычисляет **n = p * q**. Алиса вычисляет **φ(n) = (p - 1) (q - 1).** Затем она выбирает **e**, для общедоступного ключа и вычисляет **d** для частного ключа, такое, что **e * d = 1 mod φ(n)**. Алиса сохраняет **d** и публично объявляет **n** и **e**.

Подписание:

1. Алиса на основе сообщения **M** создает подпись **S**, используя секретный ключ (**d**, **n**): $\mathbf{S = M^d \bmod n}$.
2. Передает сообщение и подпись Бобу (**M**, **S**).

Проверка:

1. Боб получает **M** и **S**.
2. Применяет общедоступный ключ Алисы (**n**, **e**) к подписи, чтобы создать копию сообщения $\mathbf{M' = S^e \bmod n}$.
3. Боб сравнивает значение **M'** со значением **M**. Если два значения совпадают, Боб принимает сообщение[7].

[7] Алферов, А.П.Основы криптографии / А.П. Алферов, А.Ю. Зубов, А.С. Кузьмин, А.В. Черемушкин. – Москва, 2020. – С. 66.

2.3. Схема цифровой подписи ECDSA

История: ECDSA (Elliptic Curve Digital Signature Algorithm) - алгоритм с открытым ключом для создания цифровой подписи, аналогичный по своему строению DSA, но определённый в отличие от него не над полем целых чисел, а в группе точек эллиптической кривой.

Алгоритм ECDSA в 1999 г. был принят, как стандарт ANSI, в 2000 г. - как стандарт IEEE и NIST. Также в 1998 г. алгоритм был принят стандартом ISO.

Описание алгоритма: Стойкость алгоритма шифрования основывается на проблеме дискретного логарифма в группе точек эллиптической кривой. В отличие от проблемы простого дискретного логарифма и проблемы факторизации целого числа, не существует суб-экспоненциального алгоритма для проблемы дискретного логарифма в группе точек эллиптической кривой. По этой причине «сила на один бит ключа» существенно выше в алгоритме, который использует эллиптические кривые.

Параметры алгоритма:

1. Выбор хэш-функции **h(x)**. Для использования алгоритма необходимо, чтобы подписываемое сообщение являлось числом.

2. Выбор большого простого числа **q** - порядок одной из циклических подгрупп группы точек эллиптической кривой. Если размерность этого числа в битах меньше размерности в битах значений хэш-функции h(x) то используются только левые биты значения хэш-функции.

3. Простым числом **p** обозначается характеристика поля координат $\mathbf{F_p}$.

Генерация ключей:

1. Алиса выбирает эллиптическую кривую **Ep (a**, **b)** с простым числом **p**.

2. Алиса выбирает другое простое число **q**, чтобы использовать для вычисления.

3. Алиса выбирает секретный ключ **d**, целое число.

4. Алиса выбирает точку на кривой $\mathbf{e_1}$(.,….).

5. Алиса вычисляет $\mathbf{e_2}$ (..,....) = **d** * $\mathbf{e_1}$ (.....), другую точку на кривой.

Общедоступный ключ Алисы - (**a**, **b**, **p**, **q**, $\mathbf{e_1}$, $\mathbf{e_2}$), ее секретный ключ - **d**.

Подписание: Процесс подписания состоит главным образом из выбора секретного случайного числа, создания третьей точки на кривой, вычисления двух подписей и передачи сообщения и подписей.

1. Алиса выбирает секретное случайное число **r**, между **1** и **q - 1**.

2. Алиса выбирает третью точку на кривой **P** (**u**, **v**) = **r** * $\mathbf{e_1}$ (.....).

3. Алиса использует первые координаты **P** (**u**, **v**), чтобы вычислить первую подпись $\mathbf{S_1}$.: $\mathbf{S_1}$ = **u mod q**.

4. Алиса использует хэш сообщения, свой секретный ключ **d**, секретное случайное число **r** и $\mathbf{S_1}$, чтобы вычислить вторую подпись $\mathbf{S_2}$ = **(h(M) + d** * $\mathbf{S_1}$**) r** $^{-1}$ **mod q**,

5. Алиса передает **M**, $\mathbf{S_1}$ и $\mathbf{S_2}$.

Проверка: Процесс проверки состоит главным образом из восстановления третьей точки и подтверждения, что первая координата эквивалентна $\mathbf{S_1}$ по модулю **q**. Так как третья точка была создана подписывающим лицом, использующим секретное случайное число **r**. А верификатор не имеет этого значения. Ему нужно создать третью точку из хэша сообщения, $\mathbf{S_1}$, и $\mathbf{S_2}$.

1. Боб применяет **M**, $\mathbf{S_1}$ и $\mathbf{S_2}$ для создания двух промежуточных результатов **A** и **B**:

2. **A = h(M)**$\mathbf{S_2^{-1}}$ **mod q**.

3. **B =** $\mathbf{S_2^{-1}}$ $\mathbf{S_1}$ **mod q**.

4. Затем Боб восстанавливает третью точку **T**(**x**, **y**) = **A** * $\mathbf{e_1}$(… , …) **+ B** * $\mathbf{e_2}$(… , …)

Боб использует первую координату из **T**(**x**, **y**), чтобы проверить сообщение. Если **x =** $\mathbf{S_1}$ **mod q**, подпись принимается, иначе - отклоняется[8].

[8] Шнайер, Б. Прикладная криптография. / Б. Шнайер. – Москва, 2022. – С. 199.

2.4. Схемы с формированием подписи на основе решения системы сравнений

Генерация подписи (*R*, *S)* в схемах ЭЦП на основе сложности дискретного логарифмирования может быть осуществлена с использованием нового механизма, в котором оба элемента подписи *R и S* представляются в одинаковом виде $R=a^k \bmod p$ и $S=a^g \bmod p$, где значения *к* и *g* вычисляются одновременно как одно из решений системы из двух сравнений, записываемых в зависимости от вида проверочного соотношения. Идея этого механизма состоит в том, чтобы сделать вычислительно невозможным вычисление одного из параметров *R* или *S* при наперед заданном значении второго параметра. Параметры *R* и *S* используются как аргументы двух различных функций $F_1(R, S)$ и $F_2(R, S)$. При определенных ограничениях на значения аргументов их можно изменять таким образом, что значение функции $F_2(R, S)$ будет оставаться неизменным. При этом значение функции $F_1(R, S)$ должно изменяться таким образом, что можно подобрать пару значений *R* и *S,* при которых будет выполняться некоторое проверочное соотношение[9].

Таким образом, в определенной области пар значений *R* и *S* мы имеем $F_2 = Z = \text{const}$, поэтому проверочное соотношение в принципе может быть упрощено, что при определенном его виде позволит вычислить подпись $(R(Z), S(Z))$, зависящую от Z. При составлении конкретных вариантов подписи могут быть использованы, например, следующие пары функций (F_1, F_2): $(R/S \bmod p, RS \bmod p)$, $(RS \bmod p, R/S \bmod p)$, $(RS^M \bmod p, RS \bmod p)$, $(RS \bmod p, RS^2 \bmod p)$, $(RS^Z \bmod p, RS \bmod p)$, $(RS^Z \bmod p, R/S \bmod p)$, где $Z = R\,S \bmod p$. При этом значения *R* и *S* предполагается выражать через *к* и *g* в виде: R = a*mod p и S = mod p. Условие постоянства значения функции $F_2 = R/S \bmod p$ запишется в виде $k - g = U \bmod \gamma$, где у есть некоторый показатель,

[9] Романец, Ю.В. Защита информации в компьютерных системах и сетях. / Ю.В. Романец, П.А.Тимофеев, В.Ф. Шаньгин – М.: Радио и связь, 2021. С. 135.

к которому число a относится по модулю p, и U — случайно выбираемое число.

В случаях $F_2 = RS \bmod p$ и $F_2 = RS^M \bmod p$ условие постоянства запишется в виде $k+g = U \bmod y$ и $\kappa + gM = U \bmod \gamma$ соответственно. Учитывая, что в уравнение проверки подписи входит значение сообщения M или хэш-функции H от него, можно предложить следующие варианты проверочных соотношений:

$$R / S = y^{(RS \bmod p)^H} \alpha^{(RS \bmod p)} \bmod p,$$

$$R = Sy^{H(RS \bmod p)} \alpha^{(RS \bmod p) \bmod \delta} \bmod p,$$

$$(R / S)^{(RS \bmod p)} = y^{(RS \bmod p) \bmod \delta} \alpha^{H} \bmod p,$$

где δ есть произвольное простое число длины $|\delta| \sim 0{,}25\ |p|$. Операция $F_2 \bmod \delta$ определяет некоторую сжимающую функцию F_2' значение которой остается постоянным, если значение $F_2 = RS \bmod p$ не изменяется. Это позволяет получить и использовать в проверочном соотношении две функции, зависящие от параметров R и S, причем такие, что их значения фиксируются одновременно при условии, что параметры κ и g удовлетворяют определенным условиям.

Необходимость использования пары одновременно фиксируемых функций $F2$ и $F2$ связана с тем, что в проверочном соотношении требуется задать показатели степеней элементов γ и a, зависящие от R и S. Если это условие не выполнено, то подпись может быть легко подделана путем включения фиксированной степени γ или a как дополнительного множителя в представление одного из параметров R *и* S. Например, если проверочное соотношение имеет вид $R = Sy^{(RS \bmod p)} a^H \bmod p$, то подпись можно сформировать без использования секретного ключа. Это осуществляется следующим образом. Представим элементы подписи в виде

$$R = a^H \gamma^k \bmod p \qquad (1)$$

и

$$S = \gamma^g \bmod p \quad (2)$$

При $$\kappa + g = U \bmod \gamma \quad (3)$$

значение $Z = RS \bmod p = a^H y^U \bmod p$ является фиксированным, и мы можем обеспечить выполнимость сравнения $a^H y^k = y^g y^Z a^H \ mod\ p$ (очевидно, что при этом будет выполняться проверочное уравнение, поскольку имеет место $R = Sy^Z a^H \bmod p$), задавая еще одно условие, которому должны удовлетворять значения k и g, а именно условие

$$k = g + Z \bmod \gamma \quad (4).$$

Таким образом, следует решить систему сравнений (3) и (4) и подставить решение в качестве значений показателей степеней k и g в выражения (1) и (2), определяющие значения элементов подписи R и S[10].

Если каждый из элементов y и a входит в проверочное соотношение в степенях, зависящих от вычисляемых параметров, то подделать подпись приведенным ранее способом практически невозможно, если эти степени будут представлять собой две различные нелинейно связанные функции F_2 и F_2' (однако могут оказаться реализуемыми другие способы, которые будут рассмотрены далее). Следует подчеркнуть важность нелинейности связи между рассматриваемыми функциями. Если они будут отличаться постоянным множителем, то подделка подписи также тривиальна.

Рассмотрим схему ЭЦП с проверочным сравнением $Ry^{(RS \bmod p)} = Sa^{H(RS \bmod p)} \bmod p$. Перепишем это сравнение в виде $R \equiv S\left(\alpha^H y^{-1}\right)^{(RS \bmod p)} \bmod p$ и представим элементы подписи в виде

$$R = \left(\alpha^H y^{-1}\right)^k \bmod p \quad (4.5) \text{ и } \quad S = \left(\alpha^H y^{-1}\right)^g \bmod p \quad (6)$$

При $$k + g = U \bmod y \quad (7)$$

значение $Z = RS \bmod p = \left(\alpha^H y^{-1}\right)^U$ является фиксированным и условием выполнимости проверочного сравнения является

[10] Шнайер, Б. Прикладная криптография. Протоколы, алгоритмы, исходные тексты на языке Си. / Б. Шнайер. – М.: ТРИУМФ, 2022 – С. 408.

$$k \equiv g + Z \bmod y \qquad (8).$$

Действительно, если (7) и (8) выполняются, то имеем:

$$\left(\alpha^{H} y^{-1}\right)^{k} \equiv \left(\alpha^{H} y^{-1}\right)^{g} \left(\alpha^{H} y^{-1}\right)^{Z} \bmod p \Rightarrow R y^{(RS \bmod p)} \equiv S \alpha^{H(RS \bmod p)} \bmod p.$$

Таким образом, подпись может быть сформирована без использования секретного ключа путем совместного решения сравнений (7) и (8) и последующего вычисления элементов подписи по формулам (5) и (6).

Еще один вариант ЭЦП задается сравнением проверки подписи вида

$$(R\alpha)^{(RS \bmod p)} \equiv S^{(RS \bmod p) \bmod \delta} y^{H} \bmod p,$$

в котором внесение постоянного множителя в выражение, представляющее R или S, не позволяет сформировать подпись без знания секретного ключа, поскольку каждый элемент подписи возводится в степень, которая не является заранее заданной.

Процедура формирования подписи состоит в следующем. Предполагая, что элементы подписи будут вычисляться по формулам

$$R = a^{k} \bmod p \qquad (9)$$

и

$$S = a^{g} \bmod p \qquad (10),$$

записываем условие фиксирования показателей степеней проверочного соотношения:

$$k + g = U \bmod \gamma \qquad (11)$$

и выбираем случайное значение $U < \gamma$, которое задает конкретные значения $Z = RS \bmod p = a^{U} = \bmod p$ и $Z' = (\mathrm{RS} \bmod p) \bmod \delta = Z \bmod \delta$. Затем записываем проверочное сравнение в виде

$$\left(\alpha^{k} \alpha\right)^{Z} \equiv \left(\alpha^{g}\right)^{Z'} \left(\alpha^{x}\right)^{H} \bmod p \Rightarrow \alpha^{Z(k+1)} \equiv \alpha^{gZ' + xH} \bmod p.$$

Поскольку по предположению а относится к показателю у, то из последней формулы следует дополнительное сравнение

$$Z(k+1) \equiv gZ' + xH \bmod \gamma \qquad (12),$$

которое вместе с (11) обеспечивает выполнимость проверочного сравнения. Решая совместно сравнения (11) и (12), получаем формулы

$$g = \frac{ZU + Z - xH}{Z + Z'} \bmod \gamma \quad \text{и} \quad k = \frac{UZ' - Z + xH}{Z + Z'} \bmod \gamma,$$

по которым вычисляем значения *к* и g, а затем по (9) и (10) — элементы подписи *R* и *S.*

Описанная ранее идеология построения схем ЭЦП потенциально устраняет возможность реализации скрытых каналов со сравнительно большой пропускной способностью, поскольку по секретному ключу и подписи (*R*, *S)* вычислительно сложно найти значение *U*, которое выбирается произвольным и могло бы быть использовано для передачи блоков шифjтекста. Для вычисления *U* требуется вычислить и *к,* и g, однако для этого требуется решить задачу дискретного логарифмирования. В рассмотренных ранее схемах потенциально устраняется и возможность экзистенциальной подделки подписи, однако предстоит решить еще одну проблему, связанную с подделкой подписи на основе замены переменных.

Идея атаки на основе замены переменных состоит в том, чтобы выбрать такую пару переменных значений (вместо переменных R и S), чтобы показатели степеней в проверочном соотношении зависели только от одной переменной. Тогда эту переменную можно выбрать произвольно, а вторую переменную вычислить как неизвестную. При этом наиболее сложной операцией процедуры подделки подписи ожидается операция извлечения корней различной степени по простому модулю, которая в общем случае является достаточно легко осуществимой. Проиллюстрируем эту атаку на примере последней из рассмотренных схем. Введем новую переменную Z = RS mod *p* и переменную S выразим через Z и R: $S = R^{-1}Z \bmod p$. Теперь уравнение проверки подписи имеет вид:

$$R^Z \alpha^Z \equiv R^{-Z'} Z^{Z'} y^H \bmod p \Rightarrow R^{Z+Z'} \equiv Z^{Z'} \alpha^{-Z} y^H \bmod p,$$

где Z’ = Z mod δ. Из последнего сравнения непосредственно следует формула для вычисления значения элемента подписи R:

$$R = \left(Z^{Z'} \alpha^{-Z} y^{H}\right)^{\frac{1}{Z+Z'}} \bmod p.$$

В случаях простого и составного показателя y, к которому по модулю *p* относится число *a*, достаточно легко найти значение Z, для которого сумма Z + Z' имеет значение взаимно простое с *γ*, а значит, существует обратное по отношению к ней значение $(Z + Z')^{-1} \bmod \gamma$. Для таких значений Z легко вычисляется R, а затем и значение $S = R^{-1}Z \bmod p$[11].

Устранение такой атаки достигается следующими двумя способами:

- вместо значения R в качестве элемента подписи следует указывать значение *k*, т.е. подпись приобретает вид (k, S), при этом проверочное соотношение следует модифицировать, заменяя в нем R на $a^k \bmod p$ (фактически предлагается сделать небольшой шаг назад, поскольку при генерации подписи в гипотетической исходной схеме предварительно определялось значение к, а потом по нему вычислялся элемент подписи R);

- вместо простого модуля p может быть использован составной RSA-модуль, в результате чего без решения сложной задачи разложения модуля на простые множители операция вычисления корней будет вычислительно невыполнимой.

Возможен также вариант, в котором указанные два способа комбинируются. Во втором случае экзистенциальная подделка подписи и эффективные скрытые каналы предотвращаются. В следующих разделах эти способы рассматриваются более подробно и раскрываются некоторые дополнительные преимущества, которые они обеспечивают при синтезе конкретных схем ЭЦП.

[11] Молдовян, Н.А. Теоретический минимум и алгоритмы цифровойподписи. / Н.А. Молдовян. — СПб.: БХВ-Петербург, 2020. — 304 с.

ЗАКЛЮЧЕНИЕ

Способов защиты информации существует очень много, но каждый из них всегда можно отнести к одному из двух видов: физическое сокрытие информации от противника и шифрование информации. Зашифрованную информацию можно свободно распространять по открытым каналам связи без боязни ее раскрытия и нелегального использования. Хотя, конечно же, такая защита не абсолютно надежна, и каждый из способов шифрования характеризуется своей стойкостью, т.е. способностью противостоять криптографическим атакам.

Экономическая, политическая, юридическая и другие сферы жизни общества сегодня не могут существовать без использования цифровых подписей. Их применение находит место и в декларировании товаров и услуг, и в регистрации различных сделок по обьектам недвижимости, очень широко цифровые подписи используются в банковских системах, в электронной торговле, госзаказах, в системах обращения к органам власти, для обязательной отчётности перед гос учереждениями и для обычного юридического электронного документооборота. Целью моей работы было рассмотреть различные алгоритмы ЭЦП, как современные, так и те что уже не используются, в силу недостаточной криптостойкости. Но не смотря на то, что данные алгоритмы не используются в своём первоначальном виде они являются базой, на которой построены все действующие схемы.

В процессе написания данной курсовой работы передо мной стояли несколько задач, сделать обзор существующих алгоритмов цифровой подписи, дать определение понятию что такое ЭЦП, также основная задача рассмотреть алгоритм электронной цифровой подписи на основе решения системы сравнений. В ходе написания курсовой работы все эти задачи выполненны.

СПИСОК ЛИТЕРАТУРЫ

1. Федеральный закон от 06.04.2011 N 63-ФЗ (ред. от 19.12.2022) «Об электронной подписи» – URL: https://www.consultant.ru/document/cons_doc_LAW_112701/ (дата обращения: 20.12.2022).
2. ГОСТ Р 34.10-2001. Информационные технологии. Криптографическая защита информации. Процессы формирования и проверки цифровой подписи.
3. ГОСТ Р 34.10-94. Информационные технологии. Криптографическая защита информации. Процедуры выработки и проверки электронной цифровой подписи на базе асимметричного криптографического алгоритма
4. Семёнов, Г. Цифровая подпись. Эллиптические кривые. / Г. Семёнов. // Открытые системы. – № 7-8/2022. – С. 98.
5. Полянская, О.Ю. Инфраструктура открытых ключей. Учебное пособие. / О.Ю. Полянская, В.С. Горбатов. – Москва, 2017.
6. Алферов, А.П.Основы криптографии / А.П. Алферов, А.Ю. Зубов, А.С. Кузьмин, А.В. Черемушкин. – Москва, 2020.
7. Шнайер, Б. Прикладная криптография. / Б. Шнайер. – Москва, 2022.
8. Романец, Ю.В. Защита информации в компьютерных системах и сетях. / Ю.В. Романец, П.А.Тимофеев, В.Ф. Шаньгин – М.: Радио и связь, 2021. – 376 с.
9. Шнайер, Б. Прикладная криптография. Протоколы, алгоритмы, исходные тексты на языке Си. / Б. Шнайер. – М.: ТРИУМФ, 2022 – 816 с.
10. Игоничкина Е.В. Анализ алгоритмов электронной цифровой подписи – URL: https://security.ase.md/publ/ru/pubru86/ (дата обращения: 20.12.2022).

11. Молдовян, Н.А. Теоретический минимум и алгоритмы цифровойподписи. / Н.А. Молдовян. — СПб.: БХВ-Петербург, 2020. — 304 с.

12. Макаров В.В., Луца А.В., Стародубов Д.О. Законодательная база и алгоритмы использования электронной цифровой подписи в РФ / В.В. Макаров, А.В. Луца, Д.О. Стародубов // Международный журнал гуманитарных и естественных наук. 2020. №5-4. – URL: https://cyberleninka.ru/article/n/zakonodatelnaya-baza-i-algoritmy-ispolzovaniya-elektronnoy-tsifrovoy-podpisi-v-rf (дата обращения: 21.12.2022).

13. Анисимов, Б.В. Распознавание и цифровая обработка изображений / Б.В. Анисимов, В.Д. Курганов, В.К. Злобин. – М., 2019. – 667 с.

14. Байер, В.Н. Излучение релятивистских электронов / В.Н. Байер, В.М. Катков, В.С. Фадин. – М., 2017. – 980 с.

15. Богнер, Р. Введение в цифровую фильтрацию / Р. Богнер, А. Константинидис. – М., 2017. – 343 с.

16. Бойко Схемотехника электронных систем. Цифровые устройства / Бойко. – М.: БХВ-Петербург, 2017. – 605 с.

17. Бонч-Бруевич, А.М. Применение электронных ламп в экспериментальной физике / А.М. Бонч-Бруевич. – М.: Москва: технико-теоретической литературы; Издание 4-е, 2019. – 655 с.

18. Бурцев, В.С. Сергей Алексеевич Лебедев. К 100-летию со дня рождения основоположника отечественной электронной вычислительной техники / В.С. Бурцев. – М.: [не указано], 2017. – 184 с.