ГОСУДАРСТВЕННЫЙ КОМИТЕТ СВЯЗИ, ИНФОРМАТИЗАЦИИ И ТЕЛЕКОММУНИКАЦИОННЫХ ТЕХНОЛОГИЙ РЕСПУБЛИКИ УЗБЕКИСТАН

Ташкентский университет информационных технологий

Кафедра "Информационная безопасность"

Допустить к защите

Зав. кафедрой _____

_____

" _____ "_____ 20___ г.

Выпускная квалификационная работа

на тему: "Исследование алгоритмов электронной цифровой подписи использующихся в инфокоммуникационных системах"

Выпускник: _____ **Мамадалиев Н. К.**

Руководитель: _____ **Кучкаров Т.А.**

Рецензент: _____ **Парсиев С.С.**

Консультант по ОТ и ТБ: _____ **Кодиров Ф.М.**

Ташкент 2013 г.

ГОСУДАРСТВЕННЫЙ КОМИТЕТ СВЯЗИ, ИНФОРМАТИЗАЦИИ И ТЕЛЕКОММУНИКАЦИОННЫХ ТЕХНОЛОГИЙ РЕСПУБЛИКИ УЗБЕКИСТАН

Ташкентский университет информационных технологий

Факультет ИТ Кафедра Информационная безопасность

Направление (специальность) 5523500 - "Информационная безопасность"

"УТВЕРЖДАЮ"
Зав. кафедрой _____
_____

" __ " _____ 2013 г., протокол №__

Задание

на выпускную квалификационную работу
_____ *Мамадалиеву Нодиржону Кучкарали ўғли* _____
(фамилия, имя, отчество)

1. Тема работы: *"Исследование алгоритмов электронной цифровой подписи использующихся в инфокоммуникационных системах"* ___
2. Утверждена приказом по университету от "11" 02 2013 г. № 145-07
3. Срок сдачи законченной работы *01.06.2013 г.*
4. Исходные данные к работе *Материалы полученные из интернета, из книг, из рефератов и технических документаций*
   _____
5. Содержание расчетно-пояснительной записки (перечень подлежащих разработке вопросов) _1. *Электронная цифровая подпись*. 2. *Применение алгоритма Мак-Элис для ЭЦП. 3. Безопасность жизнедеятельности.*
6. Перечень графического материала *Материалы презентации*
   _____
7. Дата выдачи задания _____ г

Руководитель _____
(подпись)

Задание принял _____
(подпись)

8. Консультанты по отдельным разделам выпускной работы

Раздел	Ф.И.О. руководителя	Подпись, дата	
		Задание выдал	Задание получил
Обзорная и основная части	Кучкаров Т.А.		
БЖД	Кодиров Ф.М.		

9. График выполнения работы

№	Наименование раздела работы	Срок выполнения	Отметка руководителя о выполнении
1.	Электронная цифровая подпись (обзор)	29.02.13 г.	
2.	Применения алгоритма Мак-Элис для электронной цифровой подписи	20.03.13 г.	
3.	Безопасность жизнедеятельности	25.05.13 г.	
4.	Подготовка презентационного материала	30.05.13 г.	

Выпускник ________
_____ "____" _____2013 г.
(подпись)

Руководитель _____
_____ "____" _____2013 г.
(подпись)

Выпускная квалификационная работа посвящена исследованию применения алгоритмов системы кодирования Мак-Элис для электронной цифровой подписи. Показано, что в этом случае использование алгоритма Мак-Элис повышает скрытность информации. Применение параллельных кодов в алгоритме Мак-Элис, кроме расширения пространства ключей, позволяет значительно повысить криптостойкость информации, а также увеличить криптостойкость цифровой подписи по сравнению с исходным алгоритмом на основе простых линейных кодов.

Рассмотрены также вопросы безопасность жизнедеятельности.

Ушбу битирув малакавий ишда Мак-Элис кодлаш алгоритмларини электрон рақамли имзо учун қўллашни тадқиқ этилган. Мак-Элис алгоритмда параллел кодларни ишлатиш электрон рақамли имзони крипточидамлигини оширади.

Шунингдек хаёт фаолияти хавфсизлиги хам кўриб чиқилган.

Exhaust functioning is dedicated to study of the system algorithm using of the coding Mak-Elis for electronic digital signature. It Is Shown that in this case use the algorithm Mak-Elis raises the secretiveness to information. Using the parallel codes in algorithm Mak-Elis, except extensions space ключей, allows vastly to raise cryptostability information, as well as enlarge cryptostability digital signature in contrast with source algorithm on base of the simple linear codes.

They Are Considered also questions safety to vital activity.

СОДЕРЖАНИЕ

ВВЕДЕНИЕ

Развитие информационно-коммуникационных технологий (ИКТ) является одним из основных факторов благосостояния и экономического роста страны. Сегодня ИКТ становится одним из основных приоритетов государственной политики Узбекистана.

В Постановлении Президента Республики Узбекистан "О мерах по дальнейшему внедрению и развитию современных информационно-коммуникационных технологий. (Собрание законодательства Республики Узбекистан, 2012 г., № 13, ст. 139) одной из основных задачи является дальнейшее внедрение и развитие информационно-коммуникационных технологий [1].

Актуальность темы.

В настоящее время задачи криптографии выходят далеко за рамки обеспечения секретности данных [2]. По мере автоматизации процессов передачи и обработки информации, а также интенсификации информационных потоков, криптографические методы приобретают уникальное значение. Новые информационные технологии в своей основе имеют криптографию с открытым ключом (двухключевая криптография), которая позволяет реализовать протоколы, предполагающие, что секретный ключ известен только одному пользователю. Эти протоколы ориентированны на взаимное недоверие взаимодействующих сторон [2, 3].

Криптография с открытым ключом может реализовать протоколы взаимодействия сторон, которые не доверяют друг другу, данная характеристика очень важна для электронной цифровой подписи (ЭЦП), являющейся важным элементом применения криптографии. При использовании криптографии с открытым ключом ЭЦП позволяет с высокой степенью гарантии удостовериться в том, что полученное сообщение было составлено владельцем секретного ключа. Двухключевая криптография

обеспечивает строгую доказательность факта составления того или иного сообщения конкретными абонентами криптосистем. Кроме того, при использовании симметричной криптографии для ЭЦП передача секретных ключей по закрытому каналу является трудной проблемой, но данная проблема имеет простое решение для двухключевой криптографии [2].

Для системы ЭЦП возможно использование алгоритма Мак-Элис [4], который позволяет обеспечить как скрытность, так и удобство использования.

Цель работы. Разработка и обоснование модификаций алгоритма Мак-Элис в интересах повышения эффективности по скорости, скрытности и помехоустойчивости передачи информации, в частности электронной цифровой подписи.

Поставленная в работе цель включает решение следующих задач:

1. Модификация алгоритма Мак-Элис в интересах повышения скрытности, скорости и помехоустойчивости передачи информации;

2. Исследования применения комбинированных кодов в качестве линейных кодов алгоритма Мак-Элис, позволяющих значительно увеличить помехоустойчивость и скрытность передаваемой информации данного алгоритма;

3. Исследования применения модификаций алгоритма Мак-Элис для электронной цифровой подписи (ЭЦП) в радиосистемах передачи информации.

Выпускная квалификационная работа состоит из введения, трех глав и заключения.

В ***первой главе*** выпускной работы рассмотрены проблемы аутентификации данных и ЭЦП, проанализированы методы и алгоритмы ЭЦП, цифровые подписи с дополнительными функциональными свойствами, а также средства работы с электронной цифровой подписью.

Вторая глава выпускной работы посвящена применению алгоритма Мак-Элис и возможностям использования его для повышения защищенности электронной цифровой подписи, показаны его достоинства и недостатки, возможности модернизации для использования криптостойкости ЭЦП.

В ***третьей главе*** рассмотрены вопросы безопасности жизнедеятельности: анализ условий труда, вопросы организации рабочего места, а также чрезвычайные ситуации.

В ***заключении*** приведены основные результаты выполненной работы.

1. ЭЛЕКТРОННАЯ ЦИФРОВАЯ ПОДПИСЬ

1.1. Проблема аутентификации данных и электронная цифровая подпись

При обмене электронными документами по сети связи существенно снижаются затраты на обработку и хранение документов, убыстряется их поиск. Но при этом возникает проблема аутентификации автора документа и самого документа, т.е. установления подлинности автора и отсутствия изменений в полученном документе. В обычной (бумажной) информатике эти проблемы решаются за счет того, что информация в документе и рукописная подпись автора жестко связаны с физическим носителем (бумагой). В электронных документах на машинных носителях такой связи нет.

Целью аутентификации электронных документов является их защита от возможных видов злоумышленных действий, к которым относятся:

- активный перехват - нарушитель, подключившийся к сети, перехватывает документы (файлы) и изменяет их;
- маскарад - абонент *C* посылает документ абоненту *B* от имени абонента *A*;
- ренегатство - абонент *A* заявляет, что не посылал сообщения абоненту B, хотя на самом деле послал;
- подмена - абонент *B* изменяет или формирует новый документ и заявляет, что получил его от абонента *A*;
- повтор - абонент *C* повторяет ранее переданный документ, который абонент *A* посылал абоненту *B*.

Эти виды злоумышленных действий могут нанести существенный ущерб банковским и коммерческим структурам, государственным предприятиям и организациям, частным лицам, применяющим в своей деятельности компьютерные информационные технологии.

При обработке документов в электронной форме совершенно непригодны традиционные способы установления подлинности по рукописной подписи и оттиску печати на бумажном документе.

Принципиально новым решением является электронная цифровая подпись (ЭЦП).

Электронная цифровая подпись используется для аутентификации текстов, передаваемых по телекоммуникационным каналам. Функционально она аналогична обычной рукописной подписи обладает ее основными достоинствами:

- удостоверяет, что подписанный текст исходит от лица, поставившего подпись;
- не дает самому этому лицу возможности отказаться от обязательств, связанных с подписанным текстом;
- гарантирует целостность подписанного текста.

Цифровая подпись представляет собой относительно небольшое количество дополнительной цифровой информации, передаваемой вместе с подписываемым текстом.

Система ЭЦП включает две процедуры: 1) процедуру постановки подписи; 2) процедуру проверки подписи. В процедуре постановки подписи используется секретный ключ отправителя сообщения, в процедуре проверки подписи - открытый ключ отправителя.

При формировании ЭЦП отправитель прежде всего вычисляет хэш-функцию *h(M)* подписываемого текста *M*. Вычисленное значение хэш-функции *h(M)* представляет собой один короткий блок информации т, характеризующий весь текст *M* в целом. Затем число *m* шифруется секретным ключом отправителя. Получаемая при этом пара чисел представляет собой ЭЦП для данного текста М.

При проверке ЭЦП получатель сообщения снова вычисляет хэш-

функцию mh(M) принятого по каналу текста *M*, после чего при помощи открытого ключа отправителя проверяет, соответствует ли полученная подпись вычисленному значению m хэш-функции.

Принципиальным моментом в системе ЭЦП является невозможность подделки, ЭЦП пользователя без знания его секретного ключа подписывания.

В качестве подписываемого документа может быть использован любой файл. Подписанный файл создается из неподписанного путем добавления в него одной или более электронных подписей.

Каждая подпись содержит следующую информацию:

- дату подписи;
- срок окончания действия ключа данной подписи;
- информацию о лице, подписавшем файл (Ф.И.О., должность, краткое наименование фирмы);
- идентификатор подписавшего (имя открытого ключа);
- собственно цифровую подпись.

1.2. Однонаправленные хэш-функции

Хэш-функция предназначена для сжатия подписываемого документа *M* до нескольких десятков или сотен бит. Хэш-функция *h(×)* принимает в качестве аргумента сообщение (документ) M произвольной длины и возвращает хэш-значение *h(M)-H* фиксированной длины. Обычно хэшированная информация является сжатым двоичным представлением основного сообщения произвольной длины. Следует отметить, что значение хэш-функции *h(M)* сложным образом зависит от документа *M* и не позволяет восстановить сам документ *M*.

Хэш-функция должна удовлетворять целому ряду условий:

- хэш-функция должна быть чувствительна к всевозможным изменениям в тексте *M*, таким как вставки, выбросы, перестановки и т.п.;
- хэш-функция должна обладать свойством необратимости, то есть задача подбора документа *M'*, который обладал бы требуемым значением хэш-функции, должна быть вычислительно неразрешима;
- вероятность того, что значения хэш-функции двух различных документов (вне зависимости от их длин) совпадут, должна быть ничтожно мала.

Большинство хэш-функций строится на основе однонаправленной функции *f(-)*, которая образует выходное значение длиной n при задании двух входных значений длиной n. Этими входами являются блок исходного текста Mi и хэш-значение Нм предыдущего блока текста (рис. 1.1):

$$H_i = f\left(M_i, H_{i-1}\right)$$

Хэш-значение, вычисляемое при вводе последнего блока текста, становится хэш-значением всего сообщения *M*.

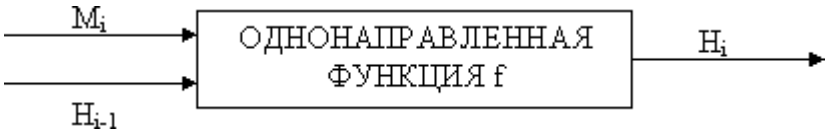

Рис. 1.1. Построение однонаправленной хэш-функции

В результате однонаправленная хэш-функция всегда формирует выход фиксированной длины *n* (независимо от длины входного текста).

1.2.1. Однонаправленные хэш-функции на основе симметричных блочных алгоритмов

Однонаправленную хэш-функцию можно построить, используя симметричный блочный алгоритм. Наиболее очевидный подход состоит в

том, чтобы шифровать сообщение M посредством блочного алгоритма в режиме CBC или CPB с помощью фиксированного ключа и некоторого вектора инициализации IV, Последний блок шифртекста можно рассматривать в качестве хэш-значения сообщения M. При таком подходе не всегда возможно построить безопасную однонаправленную хэш-функцию, но всегда можно получить код аутентификации сообщения MAC (Message Authentication Code).

Более безопасный вариант хэш-функции можно получить, используя блок сообщения в качестве ключа, предыдущее хэш-значение - в качестве входа, а текущее хэш-значение - в качестве выхода. Реальные хэш-функции проектируются еще более сложными. Длина блока обычно определяется длиной ключа, а длина хэш-значения совпадает с длиной блока.

Поскольку большинство блочных алгоритмов являются 64-битовыми, некоторые схемы хэширования проектируют так, чтобы хэш-значение имело длину, равную двойной длине блока.

Если принять, что получаемая хэш-функция корректна, безопасность схемы хэширования базируется на безопасности лежащего в ее основе блочного алгоритма. Схема хэширования, у которой длина хэш-значения равна длине блока, показана на рис. 1.2. Ее работа описывается выражениями:

$$H_0 = I_H$$
$$H_i = E_A(B)A \cdot C$$

где I_H - некоторое случайное начальное значение; A, B и C могут принимать значения $M_i, H_{i-1}, (M_i A \cdot H_{i-1})$ или быть константами.

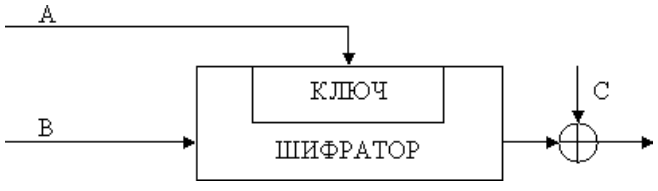

Рис. 1.2. Обобщенная схема формирования хэш-функции

Сообщение *M* разбивается на блоки *Mi* принятой длины, которые обрабатываются поочередно.

Три различные переменные *A, B* и *C* могут принимать одно из четырех возможных значений, поэтому в принципе можно получить 64 варианта общей схемы этого типа. Из них 52 варианта являются либо тривиально слабыми, либо небезопасными. Остальные 12 безопасных схем хэширования перечислены в табл.1.1.

Таблица 1.1.
Схемы безопасного хеширования, у которых длина хэш-значения равна длине блока

Номер схемы	Функция хэширования
1	$H_i = E_{H_{i-1}}(M_i) \oplus M_i$
2	$H_i = E_{H_{i-1}}(M_i \oplus H_{i-1}) \oplus M_i \oplus H_{i-1}$
3	$H_i = E_{H_{i-1}}(M_i) \oplus H_{i-1} \oplus M_i$
4	$H_i = E_{H_{i-1}}(M_i \oplus H_{i-1}) \oplus M_i$
5	$H_i = E_{M_i}(H_{i-1}) \oplus H_{i-1}$
6	$H_i = E_{M_i}(M_i \oplus H_{i-1}) M_i \oplus H_{i-1}$
7	$H_i = E_{M_i}(H_{i-1}) M_i \oplus H_{i-1}$
8	$H_i = E_{M_i}(M_i \oplus H_{i-1}) \oplus H_{i-1}$
9	$H_i = E_{M_i \oplus H_{i-1}}(M_i) \oplus M_i$
10	$H_i = E_{M_i \oplus H_{i-1}}(H_{i-1}) \oplus H_{i-1}$
11	$H_i = E_{M_i \oplus H_{i-1}}(M_i) \oplus H_{i-1}$
12	$H_i = E_{M_i \oplus H_{i-1}}(H_{i-1}) \oplus M_i$

Первые четыре схемы хэширования, являющиеся безопасными при всех атаках, приведены на рис. 1.3.

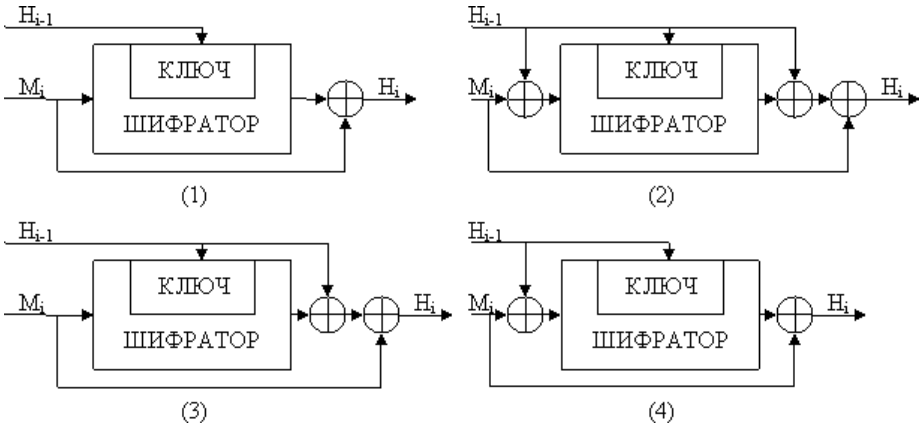

Рис. 1.3. Четыре схемы безопасного хэширования

1.2.2. Стандарт хэш-функции

ГОСТ Р 34.11-94 определяет алгоритм и процедуру вычисления хэш-функции для любых последовательностей двоичных символов, применяемых в криптографических методах обработки и защиты информации. Этот стандарт базируется на блочном алгоритме шифрования ГОСТ 28147-89. хотя в принципе можно было бы использовать и другой блочный алгоритм шифрования с 64-битовым блоком и 256-битовым ключом.

Данная хэш-функция формирует 256-битовое хэш-значение. Функция сжатия $H_i = f(M_i, H_{i-1})$ (оба операнда M_i и H_{i-1} являются 256-битовыми величинами) определяется следующим образом:

1. Генерируются 4 ключа шифрования K_j, $j = 1,...,4$, путем линейного смешивания M_i, H_{i-1} и некоторых констант C_j.

2. Каждый ключ K_j, используют для шифрования 64-битовых подслое h_i слова H_{i-1} в режиме простой замены: $S_j = E_K(h_j)$. Результирующая последовательность S_4, S_3, S_2, S_1 длиной 256 бит запоминается во временной переменной *S*.

3. Значение H_i является сложной, хотя и линейной функцией смешивания S, M_i и H_{i-1}.

При вычислении окончательного хэш-значения сообщения *M* учитываются значения трех связанных между собой переменных:

H_n - хэш-значение последнего блока сообщения;

Z - значение контрольной суммы, получаемой при сложении по модулю 2 всех блоков сообщения;

L - длина сообщения.

Эти три переменные и дополненный последний блок *M′* сообщения

объединяются в окончательное хэш-значение следующим образом:

$$H = f(ZAM', f(L, f(M', H_n))),$$

Данная хэш-функция определена стандартом ГОСТ Р 34.11-94 для использования совместно с российским стандартом электронной цифровой подписи.

1.3. Алгоритмы электронной цифровой подписи

Технология применения системы 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, как и в асимметричных системах шифрования, 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.

1.3.1. Алгоритм цифровой подписи RSA

Первой и наиболее известной во всем мире конкретной системой ЭЦП стала система RSA, математическая схема которой была разработана в 1977г. в Массачусетском технологическом институте США.

Сначала необходимо вычислить пару ключей (секретный ключ и открытый ключ). Для этого отправитель (автор) электронных документов вычисляет два больших простых числа *P* и *Q*, затем находит их произведение

$$N = P \cdot Q$$

и значение функции

$$j(N) = (P-1)(Q-1)$$

Далее отправитель вычисляет число E из условий:

$$E \cdot \xi \; j(N),\; НОД(E, j(N)) = 1$$

и число D из условий:

$$D < N, E * D \circ 1 (\mathrm{mod}\, j(N)).$$

Пара чисел *(E,N)* является открытым ключом. Эту пару чисел автор передает партнерам по переписке для проверки его цифровых подписей. Число D сохраняется автором как секретный ключ для подписывания.

Обобщенная схема формирования и проверки цифровой подписи RSA показана на рис. 1.4.

Допустим, что отправитель хочет подписать сообщение *M* перед его отправкой. Сначала сообщение *M* (блок информации, файл, таблица) сжимают с помощью хэш-функции *h(×)* в целое число m:

$$M = h(M).$$

Затем вычисляют цифровую подпись *S* под электронным документом *M*, используя хэш-значение *m* и секретный ключ *D*:

$$S = mD (\mathrm{mod}\, N)$$

Пара *(M, S)* передается партнеру-получателю как электронный

документ *M*, подписанный цифровой подписью *S*, причем подпись *S* сформирована обладателем секретного ключа *D*.

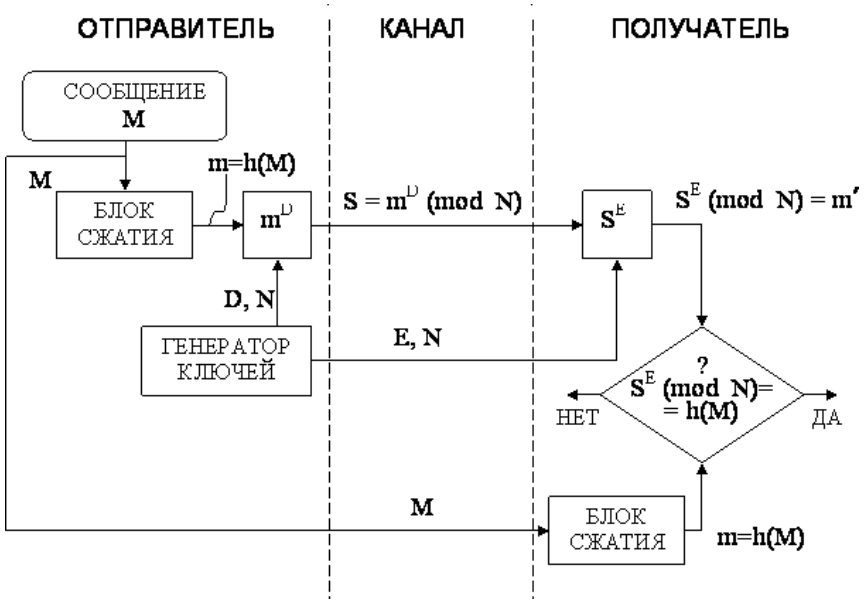

Рис. 1.4. Обобщенная схема цифровой подписи RSA.

После приема пары *(M, S)* получатель вычисляет хэш-значение сообщения *M* двумя разными способами.

Прежде всего он восстанавливает хэш-значение *m'*, применяя криптографическое преобразование подписи S с использованием открытого ключа *E*:

$$m' = S^E \pmod N$$

Кроме того, он находит результат хэширования принятого сообщения *M* с помощью такой же хэш-функции *h(×)*:

$$m = h(M)$$

Если соблюдается равенство вычисленных значений, т.е.

$$S^E \pmod N = h(m)$$

то получатель признает пару *(M, S)* подлинной. Доказано, что только обладатель секретного ключа *D* может сформировать цифровую подпись *S* по документу *M*, а определить секретное число *D* по открытому числу *E* не

легче, чем разложить модуль N на множители.

Кроме того, можно строго математически доказать, что результат проверки цифровой подписи S будет положительным только в том случае, если при вычислении S был использован секретный ключ D, соответствующий открытому ключу E. Поэтому открытый ключ E иногда называют "идентификатором" подписавшего.

Недостатки алгоритма цифровой подписи RSA

1. При вычислении модуля N, ключей E и D для системы цифровой подписи RSA необходимо проверять большое количество дополнительных условий, что сделать практически трудно. Невыполнение любого из этих условий делает возможным фальсификацию цифровой подписи со стороны того, кто обнаружит такое невыполнение. При подписании важных документов нельзя допускать такую возможность даже теоретически.

2. Для обеспечения криптостойкости цифровой подписи RSA по отношению к попыткам фальсификации на уровне, например, национального стандарта США на шифрование информации (алгоритм DES), т.е. 1018, необходимо использовать при вычислениях N, D и E целые числа не менее 2512 (или около 10154) каждое, что требует больших вычислительных затрат, превышающих на 20...30% вычислительные затраты других алгоритмов цифровой подписи при сохранении того же уровня криптостойкости.

3. Цифровая подпись RSA уязвима к так называемой мультипликативной атаке. Иначе говоря, алгоритм цифровой подписи RSA позволяет злоумышленнику без знания секретного ключа D сформировать подписи под теми документами, у которых результат хэширования можно вычислить как произведение результатов хэширования уже подписанных документов.

Пример. Допустим, что злоумышленник может сконструировать три сообщения M1, M2 и M3, у которых хэш-значения причем

$$m_1 = h(M_1),\ m_2 = h(M_2),\ m_3 = h(M_3),$$
$$m_3 = m_1 * m_2 (\mathrm{mod}\, N)$$

Допустим также, что для двух сообщений M_1 и M_2 получены законные подписи

$$S_1 = m_1^D (\mathrm{mod}\, N) \text{ и } S_2 = m_2^D (\mathrm{mod}\, N).$$

Тогда злоумышленник может легко вычислить подпись S_3 для документа M_3, даже не зная секретного ключа *D*:

$$S_3 = S_1 * S_2 (\mathrm{mod}\, N)$$

Действительно,

$$S_1 * S_2 (\mathrm{mod}\, N) = m_1^D \cdot m_2^D (\mathrm{mod}\, N) = (m_1 m_2)^D (\mathrm{mod}\, N) = m_3^D (\mathrm{mod}\, N) = S_3$$

Более надежный и удобный для реализации на персональных компьютерах алгоритм цифровой подписи был разработан в 1984 г. американцем арабского происхождения Тахером Эль Гамалем. В 1991 г. НИСТ США обосновал перед комиссией Конгресса США выбор алгоритма цифровой подписи Эль Гамаля в качестве основы для национального стандарта.

1.3.2. Алгоритм цифровой подписи Эль Гамаля (EGSA)

Название EGSA происходит от слов El Gamai Signature Algorithm (алгоритм цифровой подписи Эль Гамаля). Идея EGSA основана на том, что для обоснования практической невозможности фальсификации цифровой подписи 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 цифровой подписи RSA, связанной с возможностью подделки цифровой подписи под некоторыми сообщениями без определения секретного ключа.

Рассмотрим подробнее алгоритм цифровой подписи Эль Гамаля. Для

того чтобы генерировать пару ключей (открытый ключ секретный ключ), сначала выбирают некоторое большое простое целое число P и большое целое число *G*, причем *G<P*. Отправитель и получатель подписанного документа используют при вычислениях одинаковые большие целые числа $P\left(\approx 10^{308} \text{ или } \approx 21^{024}\right)$ и $G\left(\approx 10^{154} \text{ или } \approx 2^{512}\right)$, которые не являются секретными.

Отправитель выбирает случайное целое число *X,1<X£(P-1)*, и вычисляет

$$Y = G^X \bmod P$$

Число Y является открытым ключом, используемым для проверки подписи отправителя. Число Y открыто передается всем потенциальным получателям документов.

Число X является секретным ключом отправителя для подписывания документов и должно храниться в секрете.

Для того чтобы подписать сообщение M, сначала отправитель хэширует его с помощью хэш-функции *h(×)* в целое число *r*:

$$m = h(M),\ 1 < m < (P-1)$$

и генерирует случайное целое число *K, 1< K< (P -1)*, такое, что *K* и (P-1) являются взаимно простыми. Затем отправитель вычисляет целое число *a*:

$$a = G^K \bmod P$$

и, применяя расширенный алгоритм Евклида, вычисляет с помощью секретного ключа *X* целое число *b* из уравнения

$$m = X * a + K * b(\bmod(P-1))$$

Пара чисел *(a, b)* образует цифровую подпись *S*:

$$S = (a,b)$$

проставляемую под документом *M*.

Тройка чисел *(M, a, b)* передается получателю, в то время как пара

чисел *(X, K)* держится в секрете.

После приема подписанного сообщения *(M, a, b)* получатель должен проверить, соответствует ли подпись $S = (a,b)$ сообщению *M*. Для этого получатель сначала вычисляет по принятому сообщению *M* число

$$m = h(M),$$

т.е. хэширует принятое сообщение *M*. Затем получатель вычисляет значение

$$A = Y * ab(\mathrm{mod}\, P)$$

и признает сообщение *M* подлинным, если, и только если

$$A = G^m(\mathrm{mod}\, P).$$

Иначе говоря, получатель проверяет справедливость соотношения

$$Y^a a^b(\mathrm{mod}\, P) = G^m(\mathrm{mod}\, P)$$

Можно строго математически доказать, что последнее равенство будет выполняться тогда, и только тогда, когда подпись *S = (a, b)* под документом *M* получена с помощью именно того секретного ключа *X*, из которого был получен открытый ключ *Y*. Таким образом, можно надежно удостовериться, что отправителем сообщения *M* был обладатель именно данного секретного ключа *X*, не раскрывая при этом сам ключ, и что отправитель подписал именно этот конкретный документ *M*.

Следует отметить, что выполнение каждой подписи по методу Эль Гамаля требует нового значения *K*, причем это значение должно выбираться случайным образом. Если нарушитель раскроет когда-либо значение *K*, повторно используемое отправителем, то он сможет раскрыть секретный ключ *X* отправителя.

Пример. Выберем: числа *P=11, G=2* и секретный ключ *X=8*. Вычисляем значение открытого ключа:

$$Y = G^X \bmod P = Y = 2^8 \bmod 11 = 3$$

Предположим, что исходное сообщение M характеризуется хэш-значением *m=5*.

Для того чтобы вычислить цифровую подпись для сообщения M, имеющего хэш-значение т=5, сначала выберем случайное целое число *K=9*. Убедимся, что числа *K* и *(P-1)* являются взаимно простыми.

Действительно,

$$НОД(9,10)=1$$

Далее вычисляем элементы a и b подписи:

$$a = G^K \bmod P * 29 \bmod 11 = 6$$

элемент *b* определяем, используя расширенный алгоритм Евклида:

$$m = X * a + K * b(\bmod(P-1)).$$

При *m =5, a=6, X=8, K=9, P=11* получаем

$$5 = (6*8+9*b)(\bmod 10)$$

или

$$9*b \circ -43(\bmod 10)$$

Решение: *b = 3*. Цифровая подпись представляет собой пару: *a=6, b=3*.

Далее отправитель передает подписанное сообщение. Приняв подписанное сообщение и открытый ключ *Y=3*, получатель вычисляет хэш-значение для сообщения *M:m=5*, а затем вычисляет два числа:

$$1)\ Y^a a^b(\bmod P) = 3^6 6^3(\bmod 11) = 10(\bmod 11)\cdot$$
$$2)\ G^m(\bmod P) = 2^5(\bmod 11) = 10(\bmod 11).$$

Так как эти два целых числа равны, принятое получателем сообщение признается подлинным.

Следует отметить, что схема Эль Гамаля является характерным примером подхода, который допускает пересылку сообщения *M* в открытой форме вместе с присоединенным аутентификатором *(a, b)*. В таких случаях процедура, установления подлинности принятого сообщения состоит в проверке соответствия аутентификатора сообщению.

Схема цифровой подписи Эль Гамаля имеет ряд преимуществ по сравнению со схемой цифровой подписи RSA:

1. При заданном уровне стойкости алгоритма цифровой подписи целые числа, участвующие в вычислениях, имеют запись на 25 % короче, что уменьшает сложность вычислений почти в два раза и позволяет заметно сократить объем используемой памяти.

2. При выборе модуля *P* достаточно проверить, что это число является простым и что у числа *(P-1)* имеется большой простой множитель (т.е. всего два достаточно просто проверяемых условия).

3. Процедура формирования подписи по схеме Эль Гамаля не позволяет вычислять цифровые подписи под новыми сообщениями без знания секретного ключа (как в RSA).

Однако алгоритм цифровой подписи Эль Гамаля имеет и некоторые недостатки по сравнению со схемой подписи RSA. В частности, длина цифровой подписи получается в 1,5 раза больше, что, в свою очередь, увеличивает время ее вычисления.

1.3.3. Алгоритм цифровой подписи DSA

Алгоритм цифровой подписи DSA (Digital Signature Algorithm) предложен в 1991 г. в НИСТ США для использования в стандарте цифровой подписи DSS (Digital Signature Standard). Алгоритм DSA является развитием алгоритмов цифровой подписи Эль Гамаля и К. Шнорра.

Отправитель и получатель электронного документа используют при вычислении большие целые числа: *G* и *P* - простые числа, *L* бит каждое (512£ L £ 1024); *q* - простое число длиной 160 бит (делитель числа *(P-1))*. Числа *G*, *P*, *q* являются открытыми и могут быть общими для всех пользователей сети.

Отправитель выбирает случайное целое число *X,1<X<q*. Число *X* является секретным ключом отправителя для формирования электронной

цифровой подписи.

Затем отправитель вычисляет значение

$$Y = G^X \bmod P.$$

Число *Y* является открытым ключом для проверки подписи отправителя. Число *Y* передается всем получателям документов. Этот алгоритм также предусматривает использование односторонней функции хэширования *h(×)*. В стандарте DSS определен 3 алгоритма безопасного хэширования SHA (Secure Hash Algorithm).

Для того чтобы подписать документ *M*, отправитель хэширует его в целое хэш-значение m:

$$m = h(M), 1 < m < q$$

затем генерирует случайное целое число *K, 1<K<q*, и вычисляет число *r:*

$$r = \left(G^K \bmod P\right) \bmod q$$

Затем отправитель вычисляет с помощью секретного ключа X целое число *s*:

$$s = \frac{m + r * X}{K} \bmod q$$

Пара чисел *r* и *s* образует цифровую подпись *S = (r, s)* под документом *M*.

Таким образом, подписанное сообщение представляет собой тройку чисел *[M, r, s]*.

Получатель подписанного сообщения *[M, r, s]* проверяет выполнение условий

$$0 < r < q, 0 < s < q$$

и отвергает подпись, если хотя бы одно из этих условий не выполнено.

Затем получатель вычисляет значение

$$w = \frac{1}{s} \bmod q$$

хэш-значение

$$m = h(m)$$

и числа

$$u_1 = (m * w) \bmod q$$
$$u_2 = (r * w) \bmod q$$

Далее получатель с помощью открытого ключа Y вычисляет значение

$$v = \left(\left(G^{u^1} * Y^{u^2} \right) \bmod P \right) \bmod q$$

и проверяет выполнение условия

$$v = r$$

Если условие $v = r$ выполняется, тогда подпись $S = (r, s)$ под документом M признается получателем подлинной. Можно строго математически доказать, что последнее равенство будет выполняться тогда, и только тогда, когда подпись $S = (r, s)$ под документом M получена с помощью именно того секретного ключа X, из которого был получен открытый ключ Y. Таким образом, можно надежно удостовериться, что отправитель сообщения владеет именно данным секретным ключом X (не раскрывая при этом значения ключа X) и что отправитель подписал именно данный документ M.

По сравнению с алгоритмом цифровой подписи Эль Гамаля алгоритм DSA имеет следующие основные преимущества:

1. При любом допустимом уровне стойкости, т.е. при любой паре чисел G и P (от 512 до 1024 бит), числа q, X, r, s имеют длину по 160 бит, сокращая длину подписи до 320 бит.

2. Большинство операций с числами K, r, s, X при вычислении подписи производится по модулю числа q длиной 160 бит, что сокращает время вычисления подписи.

3. При проверке подписи большинство операций с числами u_1, u_2, v, w также производится по модулю числа q длиной 160 бит, что сокращает объем памяти и время вычисления.

Недостатком алгоритма DSA является то, что при подписывании и при проверке подписи приходится выполнять сложные операции деления по модулю q:

$$s = \frac{m + rX}{K} \bmod q \quad w = \frac{1}{s} \bmod q,$$

что не позволяет получать максимальное быстродействие.

Следует отметить, что реальное исполнение алгоритма DSA может быть ускорено с помощью выполнения предварительных вычислений. Заметим, что значение r не зависит от сообщения M и его хэш-значения m. Можно заранее создать строку случайных значений K и затем для каждого из этих значений вычислить значения r. Можно также заранее вычислить обратные значения K^{-1} для каждого из значений K. Затем, при поступлении сообщения M, можно вычислить значение s для данных значений r и K^{-1}. Эти предварительные вычисления значительно ускоряют работу алгоритма DSA.

1.4. Цифровые подписи с дополнительными функциональными свойствами

Рассматриваемые в этом разделе цифровые подписи обладают дополнительными функциональными возможностями, помимо обычных свойств аутентификации сообщения и невозможности отказа подписавшего лица от обязательств, связанных с подписанным текстом. В большинстве случаев они объединяют базовую схему цифровой подписи, например на основе алгоритма RSA, со специальным протоколом, обеспечивающим достижение тех дополнительных свойств, которыми базовая схема цифровой

подписи не обладает.

К схемам цифровой подписи с дополнительными функциональными свойствами относятся:

- схемы слепой (blind) подписи,
- схемы неоспоримой (undeniable) подписи.

1.4.1. Схемы слепой цифровой подписи

В отличие от обычных схем цифровой подписи, описанных в п.1.3, схемы слепой подписи (иногда называемые схемами подписи вслепую) являются двусторонними протоколами между отправителем A и стороной B, подписывающей документ.

Основная идея этих схем заключается в следующем. Отправитель A посылает порцию информации стороне B, которую B подписывает и возвращает A. Используя полученную подпись, сторона A может вычислить подпись стороны B на более важном для себя сообщении m. По завершении этого протокола сторона B ничего не знает ни о сообщении m, ни о подписи под этим сообщением.

Цель слепой подписи состоит в том, чтобы воспрепятствовать подписывающему лицу B ознакомиться с сообщением стороны A, которое он подписывает, и с соответствующей подписью под этим сообщением. Поэтому в дальнейшем подписанное сообщение невозможно связать со стороной A.

Приведем пример применения слепой подписи. Схема слепой подписи может найти применение в тех случаях, когда отправитель A (клиент банка) не хочет, чтобы подписывающая сторона B (банк) имела возможность в дальнейшем связать сообщение m и подпись $s_B(m)$ с определенным шагом выполненного ранее протокола.

В частности, это может быть важно при организации анонимных безналичных расчетов, когда сообщение *m* могло бы представлять денежную сумму, которую *A* хочет потратить. Когда сообщение m с подписью $s_B(m)$ предъявляется банку *B* для оплаты, банк *B* не может проследить, кто именно из клиентов предъявляет подписанный документ. Это позволяет пользователю *A* остаться анонимным.

Для построения протокола слепой подписи необходимы следующие компоненты:

1. Механизм обычной цифровой подписи для подписывающей стороны *B*. Пусть $s_B(X)$ обозначает подпись стороны *B* на документе *X*.

2. Функции *f(×)* и *g(×)* (известные только отправителю) такие, что

$$g(s_B(f_{(m)}))= s_m(m),$$

где *f(×)* - маскирующая (blinding) функция; *g(×)* - демаскирующая (unblinding) функция; *f(m)*- замаскированное (blinded) сообщение *m*.

При выборе s_B, *f* и *g* существует ряд ограничений.

Выберем в качестве алгоритма подписи s_B для стороны B схему цифровой подписи RSA (см. п.1.3) с открытым ключом (N,E) и секретным ключом D, причем *N=P * Q* - произведение двух больших случайных простых чисел.

Пусть *k* - некоторое фиксированное целое число, взаимно простое с *N*, т.е. НОД *(N, k) = 1*.

Маскирующая функция *f; Zn*® *Zn* определяется как $f(m)= m*k^E \bmod N$, а демаскирующая функция *g:Zn*-® *Zn* как $g(m)=k^{-1}m \bmod N$. При таком выборе *f, g* и *s* получаем

$$g(s_B(f(m)))= g\left(s_B\left(mk^E \bmod N\right)\right)= g\left(m^D k \bmod N\right)= m^D \bmod N = s_B(m),$$

что соответствует требованию 2.

Согласно протоколу слепой подписи, который предложил Д. Чом, отправитель *A* сначала получает подпись стороны *B* на замаскированном сообщении *m*. Используя эту подпись, сторона *A* вычисляет подпись *B* на заранее выбранном сообщении *m*, где *0£ m £ N-1*. При этом стороне *B* ничего неизвестно ни с значении *m*, ни о подписи, связанной с *m*.

Пусть сторона *B* имеет для подписи по схеме RSA открытый ключ *(N,E)* и секретный ключ *D*. Пусть *k* - случайное секретное целое число, выбранное стороной *A* и удовлетворяющее условиям *0£ m £ N-1 и НОД(N,k)*.

Протокол слепой подписи Д. Чома включает следующие шаги:

1. Отправитель *A* вычисляет замаскированное сообщение $m = mk^E \bmod N$ и посылает его стороне *B*.

2. Подписывающая сторона *B* вычисляет подпись $s = (m)^D \bmod N$ и отправляет эту подпись стороне *A*.

3. Сторона *A* вычисляет подпись $s = k^{-1}s \bmod N$, которая является подписью *B* на сообщении *m*.

Нетрудно видеть что

$$(m)^D \circ \left(mk^E\right)D \circ m^D k(\bmod N),$$

поэтому

$$k^{-1}s \circ m^D k \cdot k^{-1} \circ m^D (\bmod N) - k$$

Д. Чом разработал несколько алгоритмов слепой подписи для создания системы анонимных безналичных электронных расчетов eCash [49,108].

1.4.2. Схемы неоспоримой подписи

Неоспоримая подпись, как и обычная цифровая подпись, зависит от подписанного документа и секретного ключа. Однако в отличие от обычных цифровых подписей неоспоримая подпись не может быть верифицирована без участия лица, поставившего эту подпись. Возможно, более подходящим

названием для этих подписей было бы "подписи, не допускающие подлога".

Рассмотрим два возможных сценария применения неоспоримой подписи.

Сценарий 1. Сторона *A* (клиент) хочет получить доступ в защищенную зону, контролируемую стороной *B* (банком). Этой защищенной зоной может быть, например, депозитарий (хранилище ценностей клиентов). Сторона *B* требует от *A* поставить до предоставления клиенту доступа на заявке о допуске в защищенную зону подпись, время и дату. Если *A* применит неоспоримую подпись, тогда сторона *B* не сможет впоследствии доказать кому-либо, что *A* получил допуск без непосредственного участия *A* в процессе верификации подписи.

Сценарий 2. Предположим, что известная корпорация *A* разработала пакет программного обеспечения. Чтобы гарантировать подлинность пакета и отсутствие в нем вирусов, сторона *A* подписывает этот пакет неоспоримой подписью и продает его стороне *B*. Сторона *B* решает сделать копии этого пакета программного обеспечения и перепродать его третьей стороне *C*. При использовании стороной *A* неоспоримой подписи сторона *C* не сможет убедиться в подлинности этого пакета программного обеспечения и отсутствии в нем вирусов без участия стороны *A*.

Конечно, этот сценарий не препятствует стороне *B* поставить на пакете свою подпись, но тогда для стороны *B* будут утрачены все маркетинговые преимущества, связанные с использованием торговой марки корпорации *A*. Кроме того, будет легче раскрыть мошенническую деятельность стороны *B*.

Рассмотрим алгоритм неоспоримой цифровой подписи, разработанный Д. Чомом. Сначала опишем алгоритм генерации ключей, с помощью которого каждая сторона А. Подписывающая документ, выбирает секретный ключ и соответствующий открытый ключ.

Каждая сторона А должна выполнить следующее:

1. Выбрать случайное простое число $p = 2q + 1$, где q - также простое число.

2. Выбрать генераторное число a для подгруппы порядка q в циклической группе Zp*;

2.1.Выбрать случайный элемент b *Î* Z_p* и вычислить $a = b^{(p-1)/q} \bmod p$.

2.2. Если *a = 1*, тогда возвратиться к шагу 2.1.

3. Выбрать случайное целое *xÎ{1,2,...,q-1}* и вычислить $y = a^X \bmod p$.

4. Для стороны *A* открытый ключ равен *(p,a,y),* секретный ключ равен *x*.

Согласно алгоритму неоспоримой подписи Д. Чома, сторона А подписывает сообщение т, принадлежащее подгруппе порядка q в Zp*. Любая сторона В может проверить эту подпись при участии А.

В работе алгоритма неоспоримой подписи можно выделить два этапа:

- генерация подписи;
- верификация подписи.

На этапе генерации подписи сторона А вычисляет s=mx mod p, где s-подпись стороны А на сообщении т. Сообщение т с подписью s отсылается стороне В.

Этап верификации подписи выполняется стороной В с участием стороны А и включает следующие шаги:

1. В получает подлинный открытый ключ (p,a,y) стороны А.

2. В выбирает два случайных секретных целых числа *a, bÎ {1,2,....q-1}.*

3. В вычисляет $z = s^a y^D \bmod D$ z = sa yD mod p и отправляет значение z стороне *A*.

4. *A* вычисляет $w = z^{1/X} \bmod p$, где $xx^{-1} \circ (\bmod q)$, и отправляет

значение w стороне B.

5. B вычисляет $w' = m^a a^b \bmod p$ и признает подпись s подлинной, если и только если $w = W$.

Убедимся, что проверка подписи s работает:

$$w \cong (z)^{1/x} \cong \left(s^a y^b\right)^{1/x} \cong \left(m^{xa} \alpha^{xb}\right)^{1/x} \cong m^a \alpha^b \cong w' \bmod p.$$

Можно показать, что с высокой степенью вероятности злоумышленник не сможет заставить B принять фальшивую подпись. Предположим, что s представляет собой подделку подписи стороны A на сообщении m, т.е. $s^1 m^x \bmod p$. Тогда вероятность принятия стороной этой подписи в данном алгоритме составляет только V_q, причем эта вероятность не зависит от вычислительных ресурсов злоумышленника.

Подписавшая сторона A при некоторых обстоятельствах могла бы попытаться отказаться от своей подлинной подписи одним из трех способов:

(а) отказаться от участия в протоколе верификации;

(б) некорректно выполнить протокол верификации;

(в) объявить подпись фальшивой, даже если протокол верификации оказался успешным.

Отречение от подписи способом (а) рассматривалось бы как очевидная попытка неправомерного отказа. Против способов (б) и (в) бороться труднее, здесь требуется, специальный протокол дезавуирования. Этот протокол определяет, пытается ли подписавшая сторона A дезавуировать правильную подпись s или эта подпись является фальшивой. В этом протоколе по существу дважды применяется протокол верификации и затем производится проверка с целью убедиться, что сторона А выполняет этот протокол корректно.

Протокол дезавуирования для схемы неоспоримой подписи Д. Чома включает следующие шаги:

1. *B* принимает от стороны *A* сообщение *m* с подписью *s* и получает подлинный открытый ключ *(p,a,y)* стороны *A*.

2. *B* выбирает случайные секретные целые числа *a*, *b Î{1,2, .., q-1}*, вычисляет $z' = s^a y^b \bmod p$ и отправляет значение *z* стороне *A*,

3. А вычисляет $w' = (z')^{1/X} \bmod p$, где $xx^{-1} \circ 1(\bmod q)$, и отправляет значение *w* стороне *B*.

4. Если $w = m^a a^b \bmod p$, тогда *B* признает подпись *s* подлинной и выполнение протокола прекращается.

5. *B* выбирает случайные секретные целые числа *a`, b` Î{1,2, ...,q-1}*, вычисляет $z' = s^{a'} y^{b'} \bmod p$ и отправляет значение *t* стороне *A*.

6. *A* вычисляет $w' = (z')^{1/X} \bmod p$ и отправляет значение *w'* стороне *B*.

7. Если $w' = m^a a^b \bmod p$, тогда *B* принимает подпись s и выполнение протокола останавливается.

8. *B* вычисляет $c = \left(wa^{-b}\right)^a \bmod p$, $c' = \left(w' a^{-b}\right)^a \bmod p$. Если *c = c'*, тогда *B* заключает, что подпись *s* фальшивая; в противном случае *B* делает вывод, что подпись *s* подлинная, а сторона *A* пытается дезавуировать подпись *s*.

Нетрудно убедиться в том, что этот протокол достигает поставленной цели. Пусть *m* - сообщение и предположим, что *s* - подпись стороны *A* под сообщением *m*. Если подпись *s* фальшивая, т.е. $s' m^x \bmod p$, и если стороны *A* и *B* следуют протоколу должным образом, тогда *w = w'* (и поэтому справедливо заключение *B*, что подпись *s* фальшивая). Пусть *s* на самом деле является подписью стороны *A* под сообщением *m*, т.е. $s = m^x \bmod p$. Предположим, что *B* точно следует протоколу, а *A* не следует. Тогда вероятность того, что *w = w'* (и *A* преуспевает а дезавуировании подписи), составляет только *1/q*.

Следует отметить, что третья сторона *C* никогда не должна принимать в качестве доказательства подлинности подписи *s* запись стороной *B* протокола верификации, поскольку сторона *B* может выдумать успешную запись шага 2 и последующих шагов протокола верификации без участия подписывающей стороны *A*.

Неоспоримая подпись может быть верифицирована только путем непосредственного взаимодействия с подписывающей стороной *A*.

Разработан также алгоритм для обратимой неоспоримой подписи, которая может быть верифицирована, дезавуирована, а также преобразована в обычную цифровую подпись. Этот алгоритм основан на использовании алгоритма цифровой подписи Эль Гамаля.

1.5. Средства работы с электронной подписью

Пакет PGP

Наиболее известный - это пакет PGP (Pretty Good Privacy), – без сомнений являющийся на сегодня самым распространенным программным продуктом, позволяющим использовать современные надежные криптографические алгоритмы для защиты информации в персональных компьютерах.

К основным преимуществам данного пакета, выделяющим его среди других аналогичных продуктов следует отнести следующие:

Открытость. Исходный код всех версий программ PGP доступен в открытом виде. Любой эксперт может убедиться в том, что в программе эффективно реализованы криптоалгоритмы. Так как сам способ реализации известных алгоритмов был доступен специалистам, то открытость повлекла за собой и другое преимущество - эффективность программного кода.

Стойкость. Для реализации основных функций использованы лучшие (по крайней мере на начало 90-х) из известных алгоритмов, при этом допуская использование достаточно большой длины ключа для надежной защиты данных.

Бесплатность. Готовые базовые продукты PGP (равно как и исходные тексты программ) доступны в Интернете в частности на официальном сайте PGP.

Поддержка как централизованной (через серверы ключей) так и децентрализованной (через «сеть доверия») модели распределения открытых ключей.

Удобство программного интерфейса. PGP изначально создавалась как продукт для широкого круга пользователей, поэтому освоение основных приемов работы отнимает всего несколько часов.

Пакет GNU Privacy Guard (GnuPG)

GnuPG (www.gnupg.org) - полная и свободно распространяемая замена для пакета PGP. Этот пакет не использует патентованный алгоритм IDEA, и поэтому может быть использован без каких-нибудь ограничений. GnuPG соответсвует стандарту RFC2440 (OpenPGP).

Пакет программ Криптон «КРИПТОН»

Пакет программ Криптон (www.ancud.ru) предназначен для использования электронной цифровой подписи (ЭЦП) электронных документов.

В стандартной поставке для хранения файлов открытых ключей используются дискеты. Помимо дискет, пакет Криптон дает возможность использования всех типов ключевых носителей (смарт-карт, электронных таблеток Touch Memory и др.) [4].

2. ПРИМЕНЕНИЯ АЛГОРИТМА МАК-ЭЛИС ДЛЯ ЭЛЕКТРОННОЙ ЦИФРОВОЙ ПОДПИСИ

2.1. Алгоритм кодирования Мак-Элиса

Проанализируем, можно ли примененить алгоритм Мак-Элиса для системы электронной цифровой подписи.

Алгоритм Мак-Элис, обеспечивающий информационную скрытность передаваемой информации, основан на выборе корректирующего кода, исправляющего определенное количество ошибок, для которого существует эффективный алгоритм декодирования. С помощью закрытого ключа этот код "маскируется" под линейный код, декодирование которого не имеет эффективного решения [3]. На рис. 2.1 показан алгоритм Мак-Элис.

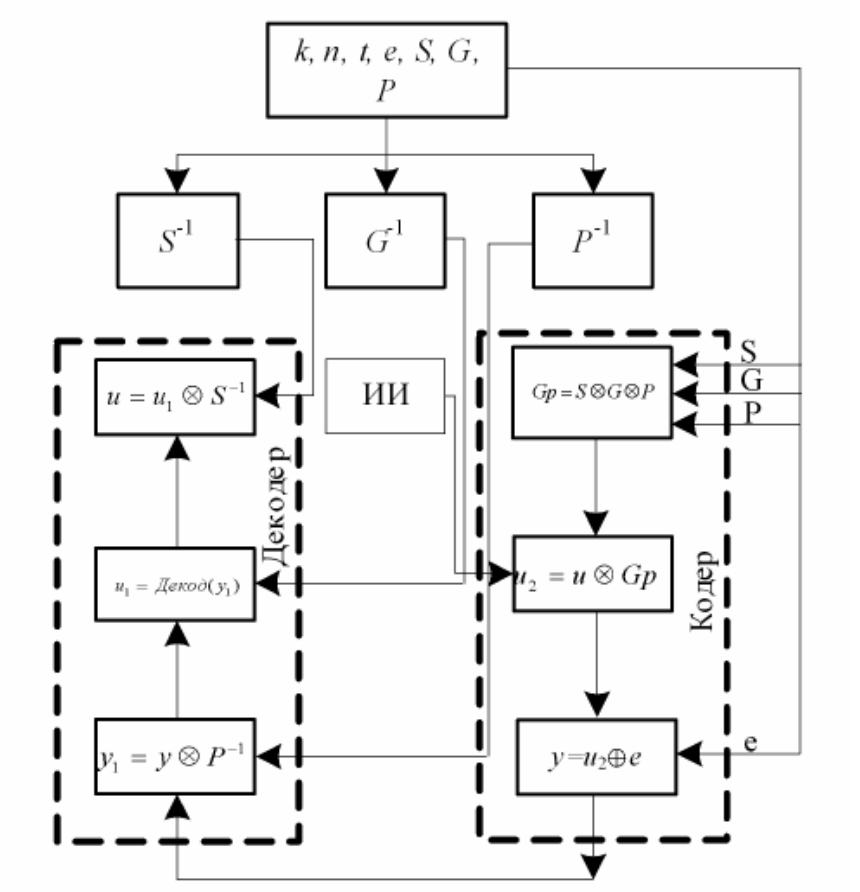

Рис. 2.1. Алгоритм Мак-Элис

Параметрами, общими для всех абонентов, являются целые числа *k*, *n* и *t*, где *k* – длина информационного сообщения, *n* – длина кодового слова, t – количество искусственно вводимых ошибок.

Каждому абоненту системы для получения открытого и соответствующего закрытого ключа следует выполнить следующие действия:

- определить порождающую матрицу $G_{k\times n}$ двоичного (*n*, *k*) - линейного кода, исправляющего t ошибок, для которого известен эффективный алгоритм декодирования;
- выбрать двоичную невырожденную матрицу $S_{k\times k}$;
- выбрать подстановочную матрицу $P_{n\times n}$;
- вычислить произведение матриц $G_p = SGP$.

Открытым ключом является пара (G_p, t), закрытым - тройка (*S*,*G*,*P*).

Для кодирования сообщения *M*, предназначенного для абонента *B*, абонент *A* должен:

- представить *M* в виде двоичного вектора *u* длины *k*;
- выбрать случайный бинарный вектор ошибок *e* длины *n*, содержащий не более *t* ошибок;
- вычислить бинарный вектор $y = G_p \oplus e$ и передать его абоненту *B*.

Получив сообщение *y*, абонент *B* вычисляет вектор $y_1 = yP^{-1}$, с помощью которого, используя алгоритм декодирования кода с порождающей матрицей *G*, получает далее векторы u_1 и $u = u_1 S^{-1}$.

Корректность приведенного алгоритма подтверждает следующее выражение [3]:

$$y_1 = yP = (uG_p \oplus e)P^{-1} = (uSGP \oplus)P^{-1} = (uS)G \oplus eP^{-1} \quad (1)$$

где eP^{-1} – вектор, содержащий не более *t* единиц. Поэтому алгоритм декодирования кода с порождающей матрицей *G* декодирует *y* в вектор

$$u_1 = uS^{-1} \ldots (2)$$

На основе исходного алгоритма кодирования информации Мак-Элис, приведенного в [3], можно предложить другой способ повышения скрытности информации и пространства ключа, основанный на использовании параллельного кода.

На рис. 2.2 предложена структурная схема данного алгоритма, содержащая две ветви; структура каждой аналогична исходному алгоритму с закрытыми - $(S_1, G_1, P_1), (S_2 G_2 P_2)$ и открытыми ключами $(t_1, G_{P_1} = S_1 G_1 P_1)$, $(t_2, G_{P_2} = S_2 G_2 P_2)$, где ИИ– источник информации, ЛК – линейный код, ГО – группообразование, ФКО – формирователь коэффициента объедения. Информация на выходе источника информации состоит из двух частей.

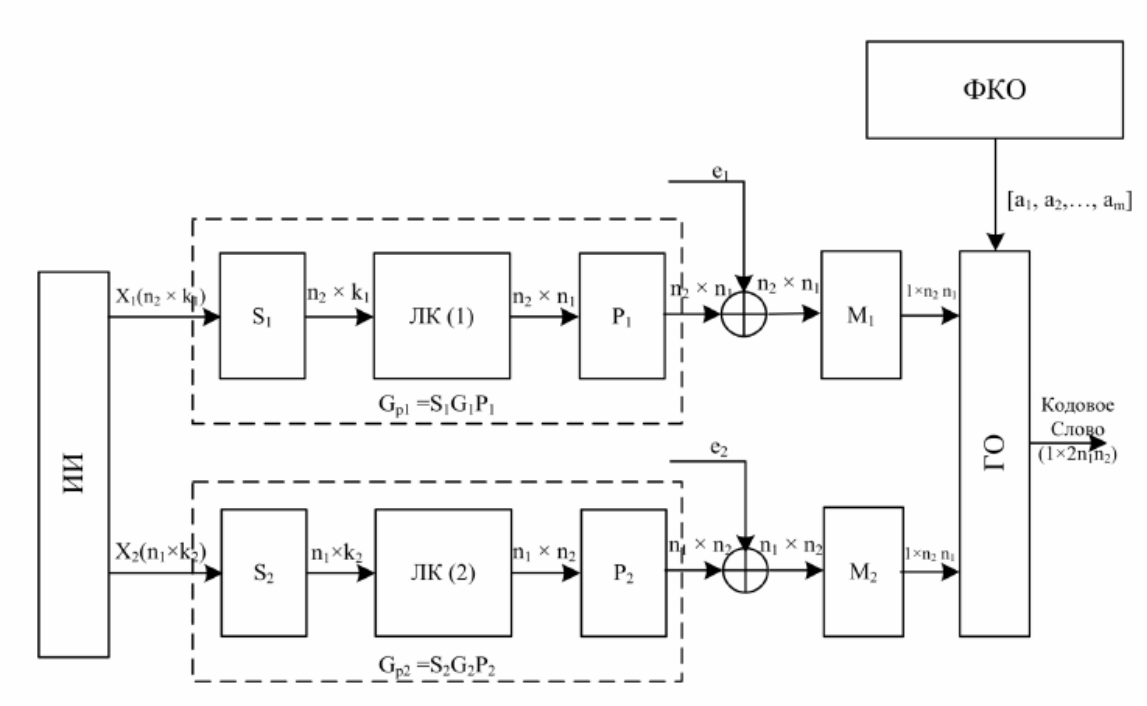

Рис. 2.2. Структурная схема данного алгоритма основанного на использовании параллельного кода

Входная информация для каждой ветви представляется строками матриц $X_{1(n_2 \times k_1)}$ и $X_{2(n_1 \times k_2)}$ Кодирование в каждой ветви выполняется по строкам матриц X_1 и X_2. Блоки M_1 и M_2 объединяют кодовые слова, поступающие из соответствующей ветви в последовательность длиной $n_1 n_2$, в чем и состоит особенность данного алгоритма. Выходная

последовательность кодера имеет длину $2n_1n_2$ Декодирование основано на разделении кодовой последовательности на две подпоследовательности длиной n_1n_2, декодирование которых осуществляется на основе исходного алгоритма.

Из анализа схемы алгоритма (рис. 2.2) следует, что объединение двух кодовых последовательностей значительно повышает скрытность передаваемой информации. При таком объединении криптостойкость информации повышается на $2^{n_1n_2}$ по сравнению с исходной схемой, изображенной на рис. 2.1. Например, при использовании в данном алгоритме двух кодов с длиной кодового слова $n=7$ выигрыш составит порядка $\sim 5\times 10^{14}$. Кроме того, при использовании закона объединения $[a_1, a_2, ..., a_m]$ расширяется пространство закрытых ключей в m! раз. Пример работы блока объединения представлен на рис. 2.3.

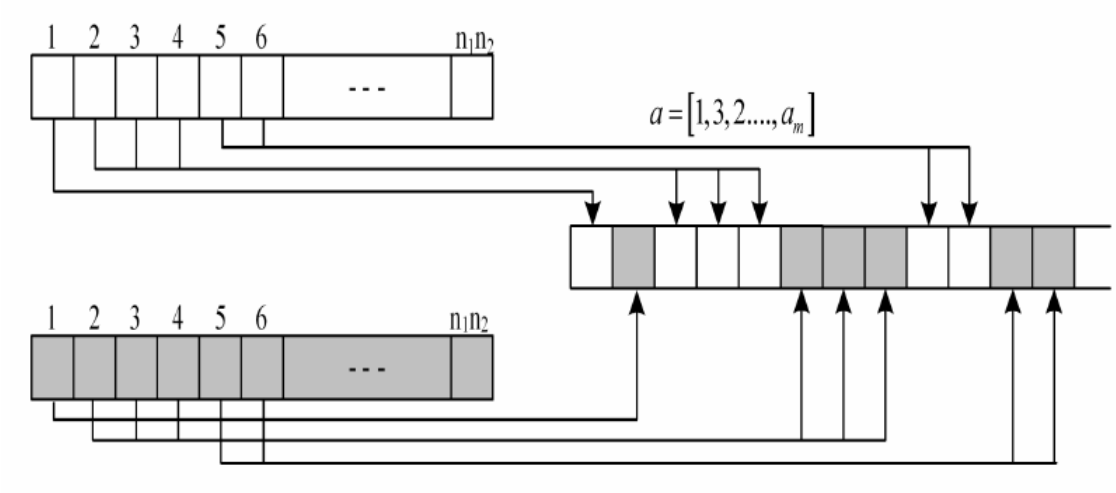

Рис. 2.3. Блок объединения

2.2. Электронная цифровая подпись с точки зрения криптоанализа

В общем случае цифровая подпись представляет собой некоторое число со специфической структурой, которое допускает проверку с помощью открытого ключа того факта, что оно было выработано для некоторого сообщения с использованием секретного ключа. Для реализации цифровой

подписи необходимо выбрать такую одностороннюю функцию с потайным ходом (с секретом) f_z, для которой при всех значениях параметра *z* область определения функции f_z, совпадает с областью её значений. При этом условии для любого сообщения, которое может быть представлено в виде числа из области определения функции $f_z(x)$, абонент *i* может сформировать с помощью секретного алгоритма число [1]

$$П = f_{z_i}^{-1}(M) \quad (3)$$

Каждый пользователь криптосистемы может по значению *П* восстановить сообщение *M*, используя открытый алгоритм шифрования E_{z_i}. Если *M* представляет собой осмысленное сообщение или может быть сопоставлено с таковым по некоторому заранее оговоренному правилу, то значение *П* может рассматриваться как цифровая подпись абонента *i* под сообщением *M*.

Реально только владелец секретного алгоритма D_{z_i} может получить "открытый" текст *П*, который с помощью алгоритма E_{z_i} зашифровывается в осмысленную криптограмму *M*, поскольку лишь абонент *i* знает способ вычисления $f_{z_i}^{-1}$

Абонент *i* может послать абоненту *j* также секретное сообщение с подписью. Для этого он зашифровывает S_i с помощью открытого алгоритма E_{z_j}, получая криптограмму [1]

$$C_i = E_{z_j}(П) \quad (4)$$

Получив зашифрованное сообщение, *j*-й абонент расшифровывает его своим секретным алгоритмом [1]

$$D_{z_j}(C_i) = П_i \quad (5)$$

затем число $П_i$ зашифровывает открытым алгоритмом *i*-го абонента [1]

$$E_{z_i}(П_i) = M_i \quad (6)$$

Таким образом, по полученной криптограмме C_i абонент *j*

восстанавливает подпись абонента *i* и исходное сообщение.

Анализ известных вариантов ЭЦП показал следующие недостатки [1, 2]:

- передача секретного алгоритма (или ключа) между абонентами в сети требует использования канала, имеющего высокую скрытность;
- требование распределения чрезмерно большого объёма ключевого материала, делает использование данных систем ЭЦП очень дорогим.

Рассмотренные ниже ЭЦП, использующие двухключевую криптографию, позволяют устранить представленные недостатки.

2.3. Исследование применения алгоритма Мак-Элис для ЭЦП

При использовании алгоритма Мак-Элис для ЭЦП абонент *i* может послать абоненту *j* секретное сообщение *M* с подписью $П_i$, для этого абоненты должны выполнить следующую работу.

1. Абонент j берет закрытые ключи - тройка двоичных матриц *(S, G, P)*.

2. Абонент *j* вычисляет открытый ключ (матрицу произведения) $G_{pj(n\times k)}=SGP$, затем он публикует пару $(G_{pj(n\times k)}, t_j)$ в справочнике открытых ключей сети.

3. Абонент *i* берет из справочника открытые ключи абонента j, выбирает случайный вектор *e*, имеющий длину *n*, вес $t \le t_j$, и затем рассчитывает кодовое слово *C* по формуле

$$C = MG_{pj} + e, \quad (7)$$

где *M* - секретное сообщение, которое будет передано *j*-му абоненту. Здесь вектор *e* играет роль цифровой подписи *i*-го абонента.

4. Получая кодовое слово *C*, *j*-й абонент выполняет декодирование с использованием матриц *(S, G, P)* (как декодирование алгоритма Мак-Элис), после этого получает секретное сообщение *M*. Таким образом, без передачи

секретного ключа между абонентами в сети *i*-й абонент послал *j*-му абоненту секретное сообщение со своей подписью, а *j*-й абонент открыл данное сообщение. Причем данный абонент может рассчитать (если он хочет проверить) цифровую подпись *i*-го абонента по формуле

$$e = C + MG_{pj}. \quad (8)$$

Таким образом, абонент j тоже может послать абоненту i секретное сообщение M_j с подписью $П_j$.

Использование алгоритма на рис. 2.2 для ЭЦП аналогично применению исходного алгоритма (рис. 2.1). Различием между ЭЦП данных алгоритмов является состав ЭЦП. При применении алгоритма на рис. 2.2 для ЭЦП подпись включает две части $П_1$ и $П_2$, соответствующие векторам ошибки e_1 и e_2. Кроме того, при использовании данного алгоритма необходима передача объединенного закона $[a_1, a_2, ..., a_m]$ между абонентами в сети по закрытому каналу.

2.4. Результаты сравнения криптостойкости модернизированного алгоритма Мак-Элиса

Результаты сравнения криптостойкости передаваемой информации исходной схеме ЭЦП и в случае использования алгоритма Мак-Элиса (рис. 2.1) и с применением кода Гоппы (1024, 524) представлены в таблице (рис. 2.2).

В таблице приняты следующие обозначения:

$К_{ИО}$ – криптостойкость информации по открытым ключам, $К_{ИЗ}$ – криптостойкость информации по закрытым ключам, $К_Э$ - криптостойкость ЭЦП, $И_Э$ – передаваемая информация для ЭЦП между абонентами, $С_{ИЭ}$ - среда передачи информации для ЭЦП между абонентами.

Таблица

Параметр \ Алгоритм	Исходная схема ЭЦП	Алгоритм рис. 2.1.	Алгоритм рис. 2.2
$K_{ИО}$	-	$1{,}37\times10^{16}$	$>1{,}37\times10^{16}$
$K_{ИЗ}$	$6{,}5\cdot10^{148}$	10^{179}	$>>10^{179}$
$K_{Э}$	$3\cdot10^{85}$	$3\cdot10^{85}$	$6\cdot10^{85}$
$И_{Э}$	Закрытый ключ (G)	Открытый ключ (G_p, t)	Открытые ключи (G_{p_1}, t_1), (G_{p_2}, t_2) и $[a_1, a_2, ..., a_m]$
$C_{ИЭ}$	Закрытый канал	Справочник	Закрытый канал и справочник

3. БЕЗОПАСНОСТЬ ЖИЗНЕДЕЯТЕЛЬНОСТИ

3.1. Личностные факторы безопасности

В ходе эволюционного и социального развития у человека выработалась и сформировалась естественная система защиты от опасностей. Эта система отличается совершенством, но имеет определенные пределы, что вызывает необходимость использования технических средств обеспечения безопасности жизнедеятельности в современной техносоциальной среде.

К личностным факторам, обуславливающим способность человека противостоять опасности, относят; психобиологический фактор, вытекающий из природных свойств психики индивида и проявляющийся в бессознательной регуляции; факторы психофизиологического качества личности; факторы социально-психологического качества личности.

Психобиологический фактор проявляется довольно сложно. Человеку, как известно, присущ целый комплекс безусловных рефлексов (инстинктов), которыми он неосознанно отвечает на различные опасности, угрожающие его организму (инстинкт самосохранения). Так при возникновении опасности повреждения закрываются глаза, отдергивается рука и т.д.

Вторым фактором, определяющим реакцию человека на опасность являются психофизиологические качества и состояния человека. Эти качества проявляются в чувствительности человека к обнаружению сигналов опасности, в его скоростных возможностях по реагированию на такие сигналы, в его эмоциональных реакциях на опасности и т.п.

Рассматривая организм человека с точки зрения воздействия на него вредных и опасных факторов в процессе деятельности, выделяют собственное тело человека и такие анализаторы, как кожный покров, органы зрения, слуха, дыхания, вкусовой чувствительности, а также

системы дыхания, мышечную, кровообращения, костную, нервную и др.Любой анализатор организма состоит из рецепторов — концевых образований нервов, которые превращают энергию раздражителя в нервные импульсы. Нервные импульсы со скоростью 120 м/с передаются в кору головного мозга. Поэтому анализаторы называют чувствующими приборами организма. Не всякий раздражитель, воздействующий на анализатор, вызывает ощущение. Чтобы оно возникло, интенсивность раздражителя должна достичь определенной величины, которую называют нижним абсолютным порогом чувствительности. Интенсивность раздражителя, после которого вызывается боль и нарушается адекватная (нормальная) деятельность анализатора, называется верхним порогом чувствительности. Минимальная разность между интенсивностями двух раздражителей, которая вызывает едва заметное различие ощущений, называется дифференциальным порогом или порогом различения. Величины порогов чувствительности нестабильны даже у одного человека. Они зависят от многих факторов, часто трудно учитываемых. Время от начала воздействия раздражителя до появления ощущения получило название латентного (скрытого) периода.

Тело человека характеризуется формой и размерами. Эти размеры называют антропометрическими характеристиками. Различают статические (соматические) и динамические характеристики. К соматическим относят размеры тела и его отдельных частей в положении стоя или сидя, к динамическим - силу костно-мышечной системы, углы вращения в суставах, изменение размеров при перемещении части тела в пространстве. Антропометрические характеристики необходимы при решении многих вопросов безопасности. С учетом этих показателей конструируют ограждения, определяют безопасные расстояния, размеры проходов, лазов, люков, а также эргонометрические требования к рычагам управления, оборудованию, кабинам водителей, рабочему месту при выполнении работ сидя или стоя.

Кожный покров тела осуществляет функции защиты от механических повреждений и участвует в терморегуляции организма. Кроме того, он выступает как анализатор, обеспечивающий восприятие прикосновения (тактильная чувствительность), вибраций, боли, тепла, холода.

Характерная особенность кожной чувствительности - **болевые ощущения**. Биологический смысл боли в том, что она является сигналом об опасности и мобилизует организм на борьбу за самосохранение. Под влиянием болевого сигнала перестраивается работа всех систем организма и повышается его реактивность.

Вкусовые и обонятельные ощущения отражают свойства веществ. В отдельных случаях они могут сигнализировать человеку о содержании в воздухе ряда веществ в количествах миллиграмма на литр. Абсолютные пороги вкусового анализатора примерно в 10 000 раз выше, чем обонятельные.

Вибрационная чувствительность, по мнению большинства следователей, обусловлена теми же рецепторами, что и тактильная. Диапазон вибрационной чувствительности находится в интервале от 1 до 12 000 Гц с наибольшей чувствительностью в диапазоне от 200 до 250 Гц.

Температурная чувствительность кожи зависит от ее температуры. Когда определенная область кожи адаптируется и становится нечувствительной к внешней температуре, то говорят, что температура среды находится на физиологическом нуле. Температура кожи несколько ниже температуры тела, равной

36-37° С. На лбу здорового человека она составляет 34-35, на лице 20-25, на животе 34, на стопах 25-270 С. Для тепловых рецепторов нижний порог чувствительности примерно равен 0,2°С, для холодовых -0,4° С. Порог различения составляет около 1°С.

Органы дыхания (легкие, дыхательные пути) человека служат для процесса газообмена. В результате кровь человека обогащается

кислородом, а наружу выделяется углекислый газ. Если вдыхаемый воздух содержит 21% кислорода, то выдыхаемый 16%. В сутки в кровь поступает до 500 л кислорода и выделяется 400 л углекислого газа.

Органы зрения и слуха наиболее активно участвуют в обеспечении безопасности. Так через орган зрения человек получает до 90% информации об окружающей среде, в том числе о ситуациина рабочем месте через средства отображения информации: табло, шкалы приборов и т. п. Значительную часть информации об окружающем мире человек получает с помощью звуковых сигналов. Слуховой аппарат человека обладает способностью различать высоту, громкость и тембр звуков, а также воспринимать положение тела в пространстве, что имеет важное значение в обеспечении безопасности.

Костно-мышечная система характеризуется прочностью костей к воздействию на них внешних сил и мышечной силой**.** Сила мышц различна и зависит от пола, состояния организма,характера выполняемых работ.

Кровообращение непременное условие жизнедеятельности организма. При остановке кровообращения смерть наступает через несколько минут, так как головной мозг весьма чувствителен к недостатку крови, а точнее - к недостатку кислорода. В организме человека находится около 5 л крови. Потеря примерно половины ее приводит к смерти. Однако свойство крови свертываться в области ран препятствует ее потере при ранениях.

Нервная система управляет всеми физиологическими функциями организма и связывает его с внешней средой. Всякого рода изменения в окружающей среде (колебания температуры, давления, состава воздуха и т. д.), воздействуя на нервные импульсы, передаются ими соответствующим органам для регулирования их деятельности, в первую очередь это касается сердечно-сосудистой системы и органов дыхания.

Характеризуя функционирование анализаторов человека и его систем, отметим, что в реальных условиях деятельности на них действуют одновременно несколько раздражителей, влияющих на всю систему анализаторов. При этом установлено, что зрительный анализатор чувствителен не только к свету, но и к действию запахов, высокой температуры, вибрации, шуму. При наличии этих факторов чувствительность зрительного аппарата снижается. Подобное явление наблюдается и у других анализаторов.

Наряду с перечисленными характеристиками для обеспечения безопасности деятельности большое значение имеет химическое состояние личности, которое зависит от состояния здоровья, степени утомления, змоционально-волевой устойчивости. Так, например, люди эмоционально неуравновешенные остро реагируют на опасные ситуации, особенно возникающие неожиданно и часто принимают ошибочные решения, допускают ошибки в предметных действиях и из-за этого попадают в несчастные случаи.

Способность человека противостоять опасности в процессе деятельности существенно зависит от третьего фактора — социально-психологических качеств личности. Социально-психологические качества личности определяются, прежде всего, отношением к выполняемой работе, мотивами деятельности, уровнем подготовки к данному виду деятельности, возраста и стажа работы.

Сегодня экспериментально доказано, что такие социально-психологические качества, как не толерантность к правилам, неуважение и недостаточная чувствительность к людям, оказываются тесно связанными с низкой защищенностью от опасностей. Установлено, что люди эгоцентрического типа (воспринимающие себя в центре мира, а остальных людей лишь как свое окружение), отличаются повышенной агрессивностьюпо отношению к другим людям, чаще остальных попадают в несчастные случаи.

Социально-психологические качества личности в значительной мере определяют поведение человека в различных сферах деятельности, в том числе его неправильные, ошибочные неосторожные действия или бездействия, отрицательно влияющие на безопасность. Анализ причин несчастных случаев, связанных с нарушением правил, норм, инструкций, не обеспечение мер безопасности связаны с недостатками человеческого фактора: недисциплинированностью, халатностью, остаточным знанием правил и норм безопасности, низкой квалификацией, малым опытом работы, необученностью безопасным методам работы, склонностью к переоценке своих возможностей и неоправданному риску, адаптацией к опасностям и т.д.

Таким образом, человек сложная саморегуляционная система, способная в зависимости от сложившейся ситуации эффективно использовать свои возможности для достижения требуемого результата и избежания при этом опасности. Если у человека, например, невысокие психофизиологические качества по противодействию опасности, то он может обеспечить безопасность за счет развития профессиональных умений и высокой мотивации к безопасному труду, т.е. за счет высокого уровня профессиональной культуры и нравственного поведения. И наоборот, человек с высокими психофизиологическими качествами даже в совокупности с профессиональными из-за низкой мотивации к безопасной деятельности может оказаться плохо защищенным от опасности.

3.2. Классификация ЧС

По причинам ЧС бывают природные, техногенные, антропогенные, экологические, социальные.

К **природным (стихийным)** ЧС относятся опасные природные явления или процессы, имеющие чрезвычайный характер и приводящие к нарушению повседневного уклада жизни более или менее значительных

групп населения, человеческим жертвам, уничтожению материальных ценностей. К ним относятся землетрясения, наводнения, цунами, извержения вулканов, селевые потоки, оползни, обвалы, ураганы и смерчи, массовые лесные и торфяные пожары, снежные заносы и лавины. К числу стихийных бедствий относятся также засухи, длительные проливные дожди, сильные устойчивые морозы, эпидемии, эпизоотии, эпифитотии, массовое распространение вредителей лесного и сельского хозяйства.

Стихийные бедствия могут происходить: в результате быстрого перемещения вещества (землетрясения, оползни); в процессе высвобождения внутриземной энергии (вулканическая деятельность, землетрясения); при повышении общего уровня рек, озер и морей (наводнения, цунами); под воздействием необычайно сильного ветра (ураганы, циклоны). Некоторые стихийные бедствия (пожары, обвалы, оползни и др.) могут возникнуть в результате действий самих людей, но последствия их всегда являются результатом действия сил природы. Для каждого стихийного бедствия характерно наличие присущих ему поражающих факторов, неблагоприятно воздействующих на состояние здоровья человека.

Стихийные бедствия являются трагедией всего государства и, особенно, для тех районов, где они возникают. В результате стихийных бедствий страдает экономика страны, так как при этом разрушаются производственные предприятия, уничтожаются материальные ценности и, самое главное, возникают потери среди людей, гибнет их жилье и имущество. Кроме того, стихийные бедствия создают крайне неблагоприятные условия для жизни населения, что может быть причиной вспышек массовых инфекционных заболеваний. Количество людей, пострадавших от стихийных бедствий, может быть весьма значительным, а характер поражений очень разнообразным. Больше всего люди страдают от наводнений (40% от общего урона), ураганов (20%),

землетрясений и засух (по 15%). Около 10% общего ущерба приходится на остальные виды стихийных бедствий.

Ряд советских и зарубежных специалистов, приводя данные о потерях при крупнейших бедствиях, предполагают, что в будущем в связи с ростом и концентрацией населения аналогичные по силе катастрофы будут сопровождаться увеличением числа жертв в десятки раз.

Техногенными ЧС принято считать внезапный выход из строя машин, механизмов и агрегатов во время их эксплуатации, сопровождающийся серьезными нарушениями производственного процесса, взрывами, образованием очагов пожаров,радиоактивным, химическим или биологическим заражениям больших территорий, групповым поражениям (гибелью) людей.К техногенным ЧС относятся аварии на промышленных объектах, строительстве, а также на железнодорожном, воздушном, автомобильном, трубопроводном и водном транспорте, в результате которых образовались пожары, разрушения гражданских и промышленных зданий, создалась опасность радиационного загрязнения, химического и бактериального заражения местности, произошло растекание нефтепродуктов и агрессивных (ядовитых) жидкостей на поверхности земли и воды и возникли другие последствия, создающие угрозу населению и окружающей среде.

Характер последствий техногенных катастроф зависит от вида аварии, ее масштабов и особенностей предприятия, на котором возникла авария (от вида транспорта и обстоятельств, при которых произошла авария).

Антропогенные ЧС являются следствием ошибочных действий персонала. Этот класс ЧС может происходить на тех же объектах, что и техногенные ЧС. Отличие состоит лишь в том, что техногенные ЧС не связаны с человеческим фактором непосредственно.

К чрезвычайным ситуациям экологического характера можно отнести: интенсивную деградацию почвы и ее загрязнение тяжелыми

металлами (кадмий, свинец, ртуть, хром и т. д.) и другими вредными веществами; загрязнение атмосферы вредными химическими веществами, шумом, электромагнитными полями; кислотные дожди; разрушение озонового слоя и т. д.

К **социальным ЧС** относятся события, происходящие в социуме (грабежи, насилия), межнациональные конфликты, сопровождающиеся применением силы; противоречия между государствами с применением оружия.

По скорости распространения опасности ЧС могут быть классифицированы на: внезапные (землетрясения, взрывы, транспортные аварии и т. д.); стремительные (пожары, гидродинамические аварии с образованием волны прорыва, аварии с выбросом газообразных СДЯВ ит. д.); умеренные (паводковые наводнения, извержения вулканов, аварии с выбросом радиоактивных, веществ); плавные с медленно распространяющейся опасностью (засухи, эпидемии, аварии на промышленных очистных сооружениях, загрязнение почвы и воды вредными химическими веществами и т. д.).

По масштабности ЧС можно подразделить на пять типов: локальные (объектовые), местные, региональные, национальные и глобальные. При локальных (объектовых) ЧС последствия ограничиваются пределами объекта народного хозяйства и могут быть устранены за счет его сил и ресурсов.

Местные ЧС имеют масштабы распространения в пределах населённого пункта, в том числе крупного города административного района, нескольких районов или области и могут быть устранены за счет сил и ресурсов области.

В региональных ЧС последствия ограничиваются пределами нескольких областей или экономического района и могут быть ликвидированы за счет сил и ресурсов республики. Национальные ЧС имеют последствия, охватывающие несколько экономических районов или

республик, но не выходящие за пределы страны. Ликвидация таких ЧС осуществляется силами и ресурсами государства, зачастую с привлечением иностранной помощи.

При глобальной ЧС ее последствия выходят за пределы страны и распространяются на другие государства. Эти последствия устраняются как силами каждого государства на своей территории, так и силами международного сообщества. Границы между всеми перечисленными типами и классами ЧС в определенной мере условны. Как уже отмечалось, некоторые стихийные бедствия - оползни, опустынивание, в отдельных случаях землетрясения, лесные и торфяные пожары и т. д. - могут иметь как чисто природное, так и природно-антропогенное происхождение. Тоже самое можно сказать и при систематизации ЧС по другим признакам.

Последствия ЧС могут быть самыми разнообразными. Они зависят от вида, характера чрезвычайной ситуации и масштаба ее распространения.

Основными видами последствий ЧС являются: гибель, заболевания людей, разрушения, радиоактивное загрязнение, химическое заражение, бактериальное заражение. Следует подчеркнуть, что на людей, находящихся в экстремальных условиях ЧС, наряду с различными поражающими факторами действуют и психотравмирующие обстоятельства, представляющие собой обычно комплекс сверхсильных раздражителей, вызывающих нарушение психической деятельности в виде так называемых реактивных (психогенных) состояний. При этом психогенное воздействие экстремальных условий складывается не только из прямой, непосредственной угрозы жизни человека, но и опосредованной, связанной с ожиданием ее реализации вне зон поражения. Если радиусы воздействия опасных и вредных факторов ЧС можно с той или иной, степенью достоверности определить заблаговременно расчет путем, то радиус психологического воздействия в реальной действительности может иметь самые различные значения. В ряде случаев он, возможно, будет во

много раз превосходить радиусы воздействия других поражающих факторов.

Территория, на которую воздействуют опасные и вредные факторы ЧС, с расположенными на ней населением, животными, зданиями и сооружениями, инженерными сетями и коммуникациями называется очагом поражения. Очаги поражения бывают простые (однородные) и сложные (комбинированные).

Простым очагом поражения называют очаг, возникший под воздействием одного поражающего фактора, например, разрушения от взрыва, пожара, только химическое или бактериальное заражение. Сложные очаги поражения возникают в результате действия нескольких поражающих факторов чрезвычайной ситуации. Например, взрыв на химическом предприятии влечет за собой разрушения, пожары, химическое заражение окружающей местности; землетрясение и ураган помимо разрушения сооружений, могут вызвать затопление прибрежной полосы, пожары от повреждения электрических сетей, химическое заражение в результате утечки СДЯВ при разрушении емкостей и т, д.

Форма очагов поражения в зависимости от природы источника опасных факторов может быть круглой — при землетрясениях, взрывах, полосной - при ураганах, смерчах, затоплениях, селевых потоках, лавинах и др., неправильной формы при пожарах, цунами, оползнях и т. п.

Независимо от происхождения и типа в развитии чрезвычайных ситуаций можно выделить четыре характерных стадии (фазы): зарождения, инициирования, кульминационную и затухания (ликвидации последствий).

На стадии зарождения складываются условия, предпосылки будущей ЧС: активизируются неблагоприятные природные процессы; накапливаются проектно-производственные дефекты сооружений и многочисленные технические неисправности; происходят сбои в работе оборудования, инженерно-технического персонала и т п.

Установить продолжительность стадии зарождения, причем весьма приблизительно, можно только с помощью регулярной статистики отказов, сбоев, «локальных» аварий, данных наблюдений сейсмических, метереологических, противоселевых и других станций.

На стадии инициирования чрезвычайного события наиболее существенно влияние человеческого фактора. Так, статистика свидетельствует, что свыше 60% аварий происходит из-за ошибок персонала.

На кульминационной **стадии** происходит высвобождение энергии или вещества, оказывающих неблагоприятное воздействие на население и окружающую среду, т. е. возникает собственно чрезвычайное событие. Особенность чрезвычайного события— цепной характер протекания, когда разрушительное действие инициирующего события многократно (иногда в сотни раз) усиливается вследствие вовлечения в процесс энергонасыщенных, токсичных, биологически активных компонентов. Образно говоря, это цепной процесс разрушительного высвобождения энергии и веществ.

Стадия затухания чрезвычайной ситуации по времени охватывает период от перекрытия (ограничения) источника опасности — локализации ЧС, до полной ликвидации ее прямых и косвенных последствий, включая всю цепочку вторичных, третичных и т. д. последствий. Продолжительность данной стадии может составлять годы, а то и десятилетия.

Заключение

1. В настоящее время широко используются такие системы кодирования, как RSA, Эль-Гамаля, и др. Однако, для реализации данных алгоритмов требуются значительные вычислительные затраты, большое количество операций, что приводит к увеличению времени шифрования и дешифрования. Высокая сложность устройств кодирования и декодирования в этих системах затрудняет реализацию алгоритмов в реальном масштабе времени и существенно ограничивает скорость передачи информации. Кроме того, эти алгоритмы имеют низкую помехоустойчивость. В отличие от этих систем алгоритм Мак-Элис, основанный на использовании линейных кодов, позволяет одновременно обеспечивать заданную помехоустойчивость и скрытность передаваемой информации.

2. Алгоритм Мак-Элис обладает такими недостатками, как низкая кодовая скорость, большая длина кодового слова, значительный размер закрытого и открытого ключа. Для широкого использования данного алгоритма, необходимо устранить вышеперечисленные недостатки.

3. Из анализа результатов использования алгоритма Мак-Элис показано, что не требуется передача секретного алгоритма (или ключа), при этом данный алгоритм обладает преимуществом по скрытности информации $\left(\approx 10^{50}\right)$.

4. Использование модифицированного алгоритма Мак-Элис позволяет значительно повысить криптостойкость информации и увеличить в два раза криптостойкость системы ЭЦП по сравнению с применением исходной схемы.

Список использованной литературы

1. Постановление Президента Республики Узбекистан "О мерах по дальнейшему внедрению и развитию современных информационно-коммуникационных технологий. (Собрание законодательства Республики Узбекистан, 2012 г., № 13, ст. 139).

2. Молдовян Н. А, Молдовян А. А. Введение в криптосистемы с открытым ключом. С-Пб, 2005. 286 с.

3. Венбо Мао. Современная криптография теория и практика. М., С-Пб. Киев, 2005. 763 с.

4. Алферов А.П. Зубов А.Ю., Кузьмин А.С., Черемушкин А.В. Основы криптографии. М.: Гелиос АРВ, 2001. 480 с.

5. Алгоритм электронной цифровой подписи на основе композиции сложностей Акбаров Д.Е., Хасанов Х.П. (ГУП «UNICON.UZ»)

6. http://www.asoiu.narod.ru/25-1.html

7. http://www.infotex72.ru/content/zakaz/bondarev.pdf

8. http://www.osp.ru/pcworld/1997/03/157204/

9. http://mind-control.wikia.com/wiki/Быстрая_цифровая_подпись

10. http://www.aladdin.kz/476.html

11. http://www.naukaspb.ru/arhiv/v_kr_kl_sod.htm

12. http://masteroid.ru/content/view/1321/49/

13. http://masteroid.ru/content/blogsection/4/15/91/91/

14. http://books.dore.ru/bs/f11bid1736.html

15. http://lpcs.math.msu.su/ver/teaching/cryptography/2008-2009-log.html

16. http://wiki-linki.ru/Citates/36427

17. http://www.cyberguru.ru/cpp-sources/algorithms/test-prostoty-rabina-page 4.html

18. http://kriptografea.narod.ru/chifpon.html

19. http://en.academic.ru/dic.nsf/enwiki/2979354

20. http://en.academic.ru/dic.nsf/enwiki/241481
21. http://ru.wikipedia.org/wiki/Схема_Шнорра
22. http://www.nestor.minsk.by/sr/2005/12/sr51218.html
23. http://www.studfiles.ru/dir/cat32/subj120/file4015/view33660.html
24. http://infoch.info/view_lesson.php
25. http://ru.wikipedia.org/wiki/ДСТУ_4145-2002
26. http://revolution.allbest.ru/programming/u00118486.html
27. http://ru.wikipedia.org/wiki/ГОСТ_Р_34.10-94
28. http://ru.wikipedia.org/wiki/Схема_Эль-Гамаля
29. http://evgenius.nightmail.ru/publ4.htm
30. http://cryptology-hr.narod.ru/lect_2_04.html
31. http://www.rohos.ru/help/crypto_algorithms.html