Последний необходимый нам факт из теории чисел: для числа e, удовлетворяющего условиям $0<e<\Phi(n)$ и НОД( $\phi(n)$,e) = 1, существует единственное число d такое, что $0<e<\Phi(n)$ и

$$de=1 \pmod{\Phi(n)} \quad (10)$$

**Однонаправленная функция РША с потайным ходом** определяется как дискретное возведение значения x в степень ключа e

$$f_z(x): y=x^e \pmod n \quad (11)$$

где x есть положительное целое число, не превосходящее n ($0<e<n$), а информация потайного хода $z=\{p,q,d\}$ , где p и q являются большими простыми числами, а значение e есть положительное целое, не превосходящее $\Phi(n)$, для которого НОД (e, $\Phi(n)$) =1

Функция $f_z(x)$ имеет обратную функцию вида

$$f^{-1}_z(y): x=y^d \pmod n \quad (12)$$

где значение d есть единственное положительное целое, меньшее $\Phi(n)$ и удовлетворяющее условию

$$de=1 \pmod{\Phi(n)}$$

Покажем, что функция $f^{-1}_z(y)$ является обратной к функции $f_z(x)$

$$(x^e)^d \pmod n = x^{ed} \pmod n = x^{\varphi(n)Q+1} \pmod n = x^{\varphi(n)Q} x \pmod n = x \pmod n$$

3) так как согласно ( 10) **de= Φ(n)Q+1** для некоторого **Q**.

**Уязвимость протокола РША (RSA) имеет три аспекта:**

**1.Имеются ограничения на выбор p и q.** Как и в протоколе Диффи-Хеллмана слабым местом разложения целых чисел на простые множители являются гладкие простые числа n. Один из наиболее эффективных алгоритмов факторизации – (p-1) алгоритм факторизации Полларда. Пусть **p**-неизвестный простой множитель числа **n** и наибольший простой множитель числа (p –1) ограничен величиной *B=Poly(k),* где k=|n|=log n, а *Poly(k)* – некоторый полином небольшой степени, зависящий от k. Число B называется границей гладкости числа (p-1). Сконструируем число $A=\Pi r^{[\log n/\log r]}$, где [log n/log r] –целая часть дроби log n/log r простые числа r<B

Очевидно, что такое число **A** делится нацело на **p-1**. Поэтому из малой теоремы Ферма следует, что $a^A \equiv 1 (\mathrm{mod}\ p)$ для любого целого числа **a**, удовлетворяющего условию НОД(a, p)=1.Если существует простой множитель **q** числа **n**, не равный множителю **p** и такой, что $a^A \neq 1 (\mathrm{mod}\ q)$, то существует целое число ***l***, не кратное числу **q**, такое, что $a^A - 1 (\mathrm{mod}\ n) = lp$. Следовательно, число НОД $(a^A - 1 (\mathrm{mod}\ n), n)$ должно быть собственным простым множителем числа **n**. Если **n=pq**, это число должно равняться числу **p**. Размер в битах числа **A** полиномиально зависит от числа **k**. Оценка длины числа **A** делается, исходя из теоремы о простом числе, по которой существует не более, чем **B/log B** простых чисел, которые меньше, чем число **B**.

Отсюда $A < B^{[\log n]B/(\log B)} < B^{kB/(\log B)}$

т.е. $|A| = \log A < kB \log 2 < k\mathrm{Poly}(k)$, длина числа **A** ограничена полиномом, зависящем от **k**. Т.е. в алгоритме РША необходимо использовать такие **p** и **q**,чтобы оценка гладкости чисел p-1и q-1

неполиномиально зависела от **|n|=log n**. Например, большие простые числа p'и q', такие, что p=2p'+1 и q=2q'+1 тоже простые. Такие простые числа называются **безопасными** простыми числами, а модули **n** алгоритма РША, имеющие два безопасных простых множителя, называются безопасными.

2. **Атака «встреча посередине» при пересылке коротких сообщений.** Небольшие исходные сообщения **m<n** могут быть восстановлены из зашифрованного текста с ненулевой вероятностью за $\sqrt{\mathbf{m}}$ попыток, т.к. факторизация в этом случае возможна и функция РША обладает мультипликативным свойством (если аргумент ф-ции РША раскладывается на некоторое количество множителей, то и значение функции будет раскладываться на такое же количество множителей:

$(m_1 * m_2)^e \equiv m_1^e * m_2^e \equiv c_1 * c_2 (\mathrm{mod}\ n)$ ). Текст, зашифрованный алгоритмом РША, трудно разложить на простые множители из-за перемешивающего свойства функции шифрования, означающего, что размер зашифрованного текста равен размеру модуля **n**.

**3.Уязвимость для атаки на основе подобранного открытого текста.**

1.Во многих криптографических протоколах стандартный режим работы – инструкция «отклик-отзыв», по которой получатель расшифровывает полученное сообщение и отсылает результат расшифровки отправителю, т.е. допускается частичный контроль блока расшифровки со стороны пользователей.

2. Криптоалгоритм должен быть таким, чтобы расшифрованный текст имеющий вид случайного набора цифр, не позволил атакующему извлечь полезную информацию.

В реальных версиях РША эти недостатки исправлены.

**Шифрование сообщений.**

**Рассмотрим использование однонаправленной функции РША с потайным ходом для шифрования сообщений, причем отправителю сообщений для этого не надо знать секретной ключевой информации.**

Выберем в качестве информации потайного хода **z = {p,q,d}**, а значения **e** и **n** открыто сообщим корреспондентам сети. Отправитель секретного сообщения **x** шифрует его с использованием несекретного ключа шифрования e согласно выражению (11). Получатель криптограммы **y** дешифрует сообщение **x** с использованием секретного ключа дешифрования **d** согласно выражению (12). Нарушитель, которому известен несекретный ключ шифрования, но неизвестна информация потайного хода **z**, не способен читать передаваемые сообщения.. Данный способ шифрования очень удобен тем, что не требует доставки секретной ключевой информации отправителям секретных сообщений.

**Цифровая подпись сообщений**

Рассмотрим использование однонаправленной функции РША с потайным ходом для обеспечения подлинности сообщений, причем получателям сообщений для этого не надо знать секретной ключевой информации.

Выберем качестве информации потайного хода **z={p,q,e}** }, а значения **d** и **n** открыто сообщим всем корреспондентам-получателям сообщений сети.

Отправитель заверяемого сообщения **x** формирует цифровую подпись у данного сообщения с использованием секретного ключа **e** формирования цифровой подписи общения согласно выражению (11).

Отправитель по каналу связи передает получателю само сообщение и его цифровую подпись. Получатель возводит полученное значение **y** цифровой подписи сообщения **x** в степень .несекретного ключа **d** проверки цифровой подписи отправителя согласно выражению (12).

Если восстановленное из цифровой подписи сообщение совпало с принятым значением сообщения, то принятое сообщение признается подлинным

Нарушитель, которому известен несекретный ключ проверки цифровой подписи отправителя, но неизвестна информация потайного хода **z**, не способен сформировать цифровую подпись произвольного сообщения, фальшивость которого не будет обнаружена получателем. Данный способ обеспечения подлинности информации; удобен тем, что проверка подлинности сообщений не требует доставки секретной ключевой информации.

Законные пользователи, знающие информацию потайного хода **z**, способны вычислительно просто определить значения прямой $f_z(x)$ и обратной $f^{-1}_z(y)$ функций.

Для нарушителя, которому неизвестна информация потайного хода **z**, определение обратной функции $f^{-1}_z(y)$ для случая шифрования или значения прямой функции $f_z(x)$ для случая обеспечения подлинности сообщений должны быть вычислительно нереализуемыми.

Рассмотрим условия, при которых обеспечивается безопасность использовать однонаправленной функции РША с потайным ходом. В общем случае нарушителю известно лишь n и e (или d). Если он способен разложить число n на множители p и q, то по известному несекретному ключу он легко вычислит секретный ключ и будет иметь полную информацию о потайном ходе z. Исследования однонаправленной функции РША с потайным ходом показали, что практически все попытки противостоящей стороны получить информацию о потайном ходе эквивалентны разложению n= p q на множители. Поэтому в последние десятилетия интенсивно исследовались методы разложения составного числа на множители. В математике такая задача называется задачей факторизации составного числа. Неизвестны доказательства на принадлежность данной задачи ни к классу P, ни к классу NP сложных задач, однако общепризнанно, что она весьма сложна. Наилучший известный алгоритм факторизации составного числа имеет субэкспоненциальную вычислительную сложность порядка

$$Cn^{\sqrt{[A \bullet \ln\ln n / \ln n}} \qquad (14)$$

где A - некоторая положительная константа, большая 1, **n = p q**
За последние годы в области разработки эффективных методов факторизации достигнуты существенные успехи, поэтому- для обеспечения требуемой безопасности применения однонаправленной функции РША с потайным ходом должны использоваться числа **p** и **q** размерностью многие сотни и даже тысячи бит. Известны примеры, как, объединив через глобальную сеть связи вычислительный ресурс сотен и тысяч ЭВМ, удается разложить на множители числа, состоящие из 130 десятичных знаков ( бит). Отметим, что сложность обращения рассматриваемой однонаправленной функции существенно уменьшается при использовании чисел **p** и **q**, не являющихся простыми. Также нежелательно использовать простые числа специального вида, например числа Мерсенна вида $2^k$-1, где k - натуральное число, для которых известны быстрые (полиномиальной сложности) алгоритмы факторизации.

**Однонаправленная функция Эль-Гамаля с потайным ходим**

На основе трудности вычисления дискретных логарифмов в алгебраическом поле можно построить однонаправленную функцию с потайным ходом.

Поле является более сложной алгебраической структурой по сравнению с группой, над его элементами можно выполнять операции сложения и умножения, а в группе - только сложение или только умножение. Например, рассмотренная ранее однонаправленная функция Диффи и Хеллмана, послужившая основой криптосистемы открытого распространения ключей, использует операции умножения над элементами группы.

В 1985 году Т. Эль-Гамаль предложил криптографическую систему цифровой подписи сообщений, которая в настоящее время считается одной из наиболее стойких криптосистем обеспечения подлинности передаваемой и хранимой информации. Рассмотрим принципы построения данной криптосистемы, послужившей основой для отечественного и американского государственных стандартов цифровой подписи сообщений

На этапе формирования ключевой информации криптосистемы равновероятно выбираются большое простое число **p** и число **g (0 <g <p )** такое, что его последовательные степени $g^0$, $g^1$,…,$g^{p-1}$ в произвольном порядке пробегают все значения от 0 до p - 1. Такое число называется примитивным элементом. Далее формирователь ключа случайным образом выбирает целое число **x**, удовлетворяющее условию 0<g<(p-1) и вычисляет значения $y=g^x(\mathrm{mod}\ p)$
Число **x** является ключом формирования цифровой подписи сообщений отправителя и должно храниться корреспондентом-отправителем сообщений в секрете, а значение y сообщается всем как открытый ключ проверки цифровой подписи сообщений отправителя. Так же открыто всем корреспондентам сети с выполнением мер обеспечения их подлинности сообщаются значения параметров криптосистемы p,g и q.
Чтобы заверить цифровой подписью передаваемое сообщение M, двоичное представление которого должно быть меньше значения p: 0<M<p, (16)

Отправитель равновероятным недетерминированным образом выбирает случайное число k ( 0<k<(p – 1)) так, чтобы числа k и p- 1 не имели общих делителей, кроме 1, то есть их наибольший общий делитель НОД (k, p-1)=1. (17)

Далее отправитель вычисляет значение первого параметра цифровой подписи сообщения:

$$r = g^{k} (\mathrm{mod}\ p) \qquad (18)$$

составляет уравнение вида

$$M=xr + ks \bmod(p-1), \qquad (19)$$

и решает его относительно второго параметра цифровой подписи сообщения

$$s = k^{-1}(M - xr) \bmod(p-1) \qquad (20)$$

где $k^{-1}$ есть число, обратное к числу k по mod (p - 1).

Цифровой подписью сообщения M является пара чисел r и s. Отправитель сообщения по каналу связи передает получателю само сообщение M и его цифровую подпись { r, s}. Отметим, что значения x и k сохраняются отправителем сообщения в тайне от всех (после формирования цифровой подписи сообщения использованное значение k целесообразно гарантированно стереть для обеспечения безопасности использования криптосистемы).

На приеме корреспондент-получатель сначала проверяет допустимость принятого значения r^.

Если оно не находится в пределах $0 < r^ < p - 1$, принятое сообщение отвергается как ложное. Если принятое значение s^ не находится в пределах $0 < s^ < p - 1$, сообщение также отвергается как ложное.

Получатель удостоверяется в подлинности принятого сообщения m^ заверенного принятой подписью {r^ , s^}, и отсутствии в нем искажений тогда и только тогда, когда выполняется равенство:

$$g^{\hat{m}} (\mathrm{mod}\, p) = y^{\hat{r}} (\hat{r})^{s} (\mathrm{mod}\, p) \qquad (21)$$

Рассмотрим пример формирования и проверки цифровой подписи сообщения в системе Эль-Гамаля. Для наглядности размерность параметров криптосистемы выбрана малой, что недопустимо при использовании практических средств криптографической защиты информации.

Пример: пусть на этапе генерирования ключей выбран модуль криптосистемы p = 1 1 и примитивный элемент g = 2 поля Галуа GF(11). Проверим правильность выбора p и g, вычислив последовательные значения выражения $g^i \pmod p$ для 1.<i<p. Анализируя эти значения, убедимся в их неповторяемости:

$2^1 \pmod{11}=2$ $2^6 \pmod{11}=9$

$2^2 \pmod{11}=4$ $2^7 \pmod{11}=7$

$2^3 \pmod{11}=8$ $2^8 \pmod{11}=3$

$2^4 \pmod{11}=5$ $2^9 \pmod{11}=6$

$2^5 \pmod{11}=10$ $2^{10} \pmod{11}=1$

Выберем секретный ключ формирования цифровой подписи сообщений отправителя x = 8 (1< x <p-1) и вычислим открытый ключ проверки цифровой подписи:

$$g^x (\mathrm{mod}\ p) = 2^8 \ \mathrm{mod} 11 = 3$$

Пусть необходимо подписать сообщение M = 5. Выберем случайное число k= 9 (0 < k <p-1) и убедимся, что НОД (k, p -1) = 1. Действительно, НОД(9,10)=1. Вычислим

$$r = g^k (\mathrm{mod}\ p) = 2^9 \ \mathrm{mod} 11 = 6$$

Составим уравнение вида (19):

$M=xr + ks \bmod(p-1)$ и решим его: $s=k^{-1}(M-xr)\bmod(p-1)=3$

Итак, для сообщения M = 5 его цифровой подписью является пара чисел г = 6, s=3. При отсутствии воздействия нарушителя и ошибок канала связи $M^{\wedge} = M; r^{\wedge} = r; s^{\wedge} = s$

Получатель сообщения вычисляет $g^{m\wedge} (\mathrm{mod}\ p)=2^5 \mathrm{mod} 11=10$

$y^{r\wedge} (r\wedge)^s (\mathrm{mod}\ p)=3^6 * 6^3\ \mathrm{mod} 11=10$

Таким образом, принятое сообщение не искажено и получено от законного корреспондента-отправителя.

**Условия обеспечения безопасности криптографической системы цифровой подписи сообщений Эль-Гамаля при активных атаках нарушителя** Если противоборствующая сторона, зная открытый ключ проверки цифровой подписи сообщений, способна вычислить секретный ключ **x** из уравнения (15), то это означает полный взлом криптосистемы. Вычислительная сложность нахождения ключа формирования цифровой подписи сообщений для рассматриваемой криптосистемы соответствует вычислительной сложности нахождения ключа дешифрования системы шифрования Эль-Гамаля. Обеспечение требуемой стойкости криптосистемы цифровой подписи сообщений Эль-Гамаля требует выбора параметра **n** двоичной длины не менее 768 бит (1024 бита для перспективных систем аутентификации информации).

Для данной криптосистемы существует еще один способ вычисления нарушителем секретного ключа формирования цифровой подписи сообщений. Пусть противостоящая сторона знает несколько открытых сообщений $M_t$ и их подписи $\{r^i, s^i\}$ для $i = 1, ..., n$. Для вычисления секретного ключа **x** формирования цифровой подписи, что означает при успехе полный взлом крипто-системы, нарушитель может построить систему из **n** линейных уравнений с **n+1** неизвестными $k^1, k^2, \ldots, k^n, x$

Такая система неразрешима и нарушитель не способен однозначно определить ключ **x**. Однако если хотя бы один раз случайные числа **k** повторяются, число неизвестных не будет превышать числа уравнений и система имеет решение. Система уравнений также будет иметь решение, если случайные числа **k** зависимы друг от друга и любое из них можно выразить через остальные. Поэтому для сохранения в тайне ключа x необходимо строго выполнять условие неповторимости и независимости случайных чисел **k.**

Существует также опасность подделки нарушителем сообщений и их подписей без знания ключа формирования подписи. Пусть нарушитель знает хотя бы одно сообщение M и его подпись {r,s}. На практике это условие обычно выполняется всегда. Нарушитель подыскивает такую пару чисел (u, w). что выполняется условие

НОД (w, p-l=)11.

Далее он вычисляет ложную цифровую подпись {r*, s*}, действуя следующим образом:

$$R^* = g^u y^w (\mathrm{mod}\, p) = g^{u+xw} (\mathrm{mod}\, p)$$

$$S^* = -r \cdot w^{-1} \mathrm{mod}(p-1)$$

Противостоящая сторона затем формирует из истинного сообщения ложное сообщение M* = s • n mod (p - 1).

Получатель такого ложного сообщения не в состоянии выявить факт подделки, так как выполняется условие проверки:

$$(g^{m} \cdot g^{-rx})^{-s1} = g^{u} \cdot y^{w} = r^{*}$$

где s1 есть обратный элемент к **s**.

Для защиты от этой атаки в сообщение M необходимо включать формируемую специальным образом избыточность, которую легко можно было бы проверить получателю сообщений. Для этого в криптосистеме Эль-Гамаля можно, например, использовать формирование избыточности из самого заверяемого сообщения, как предлагается в стандарте ISO / IEC 9796

Другим, более эффективным средством для исключения возможности подделки сообщений и их цифровых подписей является **хэширование** самого сообщения с использованием устойчивых к коллизиям криптографических хэш-функций. В этом случае параметр s цифровой подписи вычисляется от хэшированного сообщения **h (n)** и выражение (20) можно переписать в виде:

И тогда нарушитель или недобросовестный получатель сообщения не в состоянии фальсифицировать сообщение **n** и его цифровую подпись **{r, s}** так, чтобы попытка обмана не была обнаружена. Поэтому получатель сообщения сначала должен вычислить его хэшированное значение h (M^) и вместо выражения (21) проверять тождество выражения $g^{h(M^)} = y^{-r^} r^{s^} \pmod p$ (27), где {r^, s^} - принятые значения цифровой подписи сообщения M^.

**Уязвимость системы цифровой подписи Эль-Гамаля**

**1. Атака Блайхенбахера.**

Получатель должен выбрать случайный элемент $g \in F^*_p$. Если системные пользователи обладают одними и теми же открытыми параметрами **g** и **p**, то необходимо проверить, насколько случайным является параметр **g**. Злоумышленник генерирует $g = \beta^t \pmod p$, где $\beta = cq$ и $c < b$. Вычисление дискретного логарифма $y^q$ по основанию $g^q$ не создает трудностей $z \equiv x \pmod b$, далее вычисляется $r \leftarrow \beta = cq$, $s \leftarrow t(m - cqz) \pmod{p-1}$

Атаку можно предотвратить, если во время верификации проверять условие, что **r** не делится на **q**

**2. Предостережение относительно выбора случайного параметра k**

Генерация цифровой подписи Эль-Гамаля – вероятностный алгоритм, т.к. **k** –случайное. **k** используется один раз, т.е. шифрование закрытого ключа **x** идет 1 раз. Важно, чтобы число **k** никогда не использовалось для других сообщений. (при повторном использовании **k** нарушитель может построить систему из **n** линейных уравнений с **n+1** неизвестными $k^1$, $k^2$, …, $k^n$, x, и система решается).

3**. Предотвращение экзистенциальной подделки подписи.**

Алгоритмы составления подписи и ее проверки образуют пару ОНФ с секретом. Т.к. функция проверки ЦП вычисляется в направлении зашифрованный текст s –исходное сообщение m, то схемы ЦП, основанные на ОНФ с секретом, позволяют подделывать правильные пары «сообщение-подпись», применяя алгоритм проверки в направлении от s к m. Но, благодаря свойству перемешивания, которым он обладает, сообщение выглядит бессмысленным.

Такой способ подделки называется **экзистенциальным**. Для предотвращения подделки достаточно включать в исходное сообщение избыточность, позволяющую выявить подлог. При хэшировании **экзистенциальная атака** становится невозможной**.**

**Система шифрования информации эль-Гамаля** относится к классу несимметричных криптосистем и предназначена для шифрования сообщений открытым ключом и дешифрирования секретным ключом, известным только получателю.

1.На этапе генерирования ключей шифрования центр формирования ключей (ЦФК) выбирает большое простое число **p** и первообразный элемент **g** поля $\mathbf{F_p}$, удовлетворяющего условию $\mathbf{1<g<p.}$

2.ЦФК выбирает случайное $\mathbf{x}$ $\mathbf{(1<x\leq p-2)}$ и вычисляет открытый ключ **y** $\mathbf{y=g^x (mod\ p)}$**. y,p,g** известны всем.

3.На этапе шифрования сообщение M представляется в виде двоичного числа длиной не более |p| бит($0<\log M< |p|$)

Если двоичное представление сообщения превышает p, то оно разделяется на несколько $M_i$ длиной по | p| бит, которые шифруются последовательно.

4. Отправитель A случайным и равновероятным для каждого сообщения образом выбираем целое k $1< k \leq p-2$

5. Для шифрования очередного блока M отправитель вычисляет криптограмму, состоящую из двух чисел

$C_1=g^k \pmod p$ и $C_2=M\, y^k \pmod p$ и передает их получателю по открытому каналу

6. Получатель вычисляет $C_1^{-x} \pmod p$, используя **x**, а затем

$M=C_2(C_1^{-x}) \pmod p$

$M=C_2(C_1^{-x}) \pmod p = C_2(g^k)^{-x} \pmod p = My^k g^{-kx} \pmod p =$

$= M(g^x)^k g^{-kx} \pmod p = M$ $p \geq 1024$