ТРЕБОВАНИЯ К КРИПТОГРАФИЧЕСКИМ СИСТЕМАМ

В связи с наступлением информационной эпохи, любая информация становится все более и более ценным ресурсом, и товаром. Создание быстрых глобальных коммуникаций и распространение различных информационных сетей, а также революция в сфере разработки мощных средств информационной обработки, доступных широким массам, значительным образом изменили форму современного общества. Различные сферы экономической деятельности тесно переплелись с компьютерными сетями. Для защиты интересов организаций и их клиентов следует вводить эффективные меры безопасности. Ко всему прочему, переосмысление роли информации привело к появлению нового вида войн – информационных. Поэтому неудивительно, что в настоящее время появилась острая потребность в качественном шифровании передаваемой и хранящейся информации. Процесс зашифровывания различных сообщений начинает играть все большую роль в повседневной жизни. Сегодня наши телефонные разговоры передаются по спутниковым каналам, а наши электронные сообщения проходят через различные компьютерные системы, и можно осуществить перехват передаваемой информации по обоим типам связи, что ставит под угрозу нашу частную жизнь.

Актуальность поднимаемой проблемы неразрывно связана с проблемами защиты частной жизни и персональных данных, особенно в сети. Как одним из немногих способов решения, является использование одного из популярных алгоритмов шифрования. Однако стоит заметить, что использование наиболее стойких, защищенных от атак, вариаций алгоритмов ведет к увеличению времени шифрования и, соответственно, замедлению работы систем. Соответственно, возникает дилемма о том, что же предпочесть: скорость работы или надежность защиты обрабатываемых данных. Ключом к решению является поиск баланса. Но существуют системы и ситуации, в которых баланс между скоростью и надежностью невозможен, так как оба являются критичными параметрами. К примеру, необходимо обеспечить высокое быстродействие при

заданных критериях защиты. И здесь возможным решением может стать создание быстрых и надежных криптосистем, потребность в которых, с все более увеличивающимся объемом обрабатываемой информации, с каждым годом неуклонно растет. Итак, целью является разработка криптосистемы, которая давала бы высокие показатели безопасности и быстродействия.

Процесс криптографического закрытия данных может осуществляться как программно, так и аппаратно. Аппаратная реализация отличается существенно большей стоимостью, однако ей присущи и преимущества: высокая производительность, простота, защищенность и т.п. Программная реализация более практична, допускает определенную гибкость в использовании.

Независимо от способа реализации для современных криптографических систем сформулированы следующие общепринятые требования:

1. Принцип Керкгоффса, который гласит, что знание алгоритма шифрования не должно снижать криптостойкости шифра. Это основополагающее требование, сформулированное в XIX веке, подразделяет криптосистемы на системы общего использования, т.е. алгоритм может быть доступен потенциальному нарушителю, и ограниченного использования, т.е. алгоритм шифрования держится в тайне. Данный принцип должен выполняться во всех массово используемых криптосистемах. Самыми яркими примерами последствий несоблюдения этого требования могут служить взлом шифров в системе сотовой связи GSM или защите дисков DVD от незаконного воспроизведения.

2. Зашифрованное сообщение должно поддаваться чтению только при наличии ключа. Используемое в программе MSWord6.0/95 “шифрование” документа на самом деле только запрещало его открытие в данной программе. Сам же текст не шифровался и был доступен для чтения в любом текстовом редакторе.

3. Шифр должен быть стойким даже в случае если нарушителю известно достаточно большое количество исходных данных и соответствующих им зашифрованных данных.

4. Число операций, необходимых для расшифровывания информации путем перебора всевозможных ключей должно иметь строгую нижнюю оценку и должно либо выходить за пределы возможностей современных компьютеров, с учетом возможности организации сетевых вычислений, или требовать создания дорогостоящих вычислительных систем.

5. Незначительное изменение ключа или исходного текста должно приводить к существенному изменению вида зашифрованного текста. Надо сказать, что данное требование не выполняется по отношению практически ко всем шифрам донаучной криптографии.

6. Структурные элементы алгоритма шифрования должны быть неизменными.

7. Длина шифрованного текста должна быть раной длине исходного текста.

8. Дополнительные биты, вводимые в сообщение в процессе шифрования, должны быть полностью и надежно скрыты в шифрованном тексте.

9. Не должно быть простых и легко устанавливаемых зависимостей между ключами, последовательно используемых в процессе шифрования.

10. Любой ключ из множества возможных должен обеспечивать равную криптостойкость. В этом случае принято говорить о линейном, однородном, пространстве ключей.

ОСНОВНЫЕ СВЕДЕНИЯ О КРИПТОАНАЛИЗЕ

Главным действующим лицом в криптоанализе выступает нарушитель или криптоаналитик. Под ним понимают лицо или группу лиц, целью которых является прочтение или подделка защищенных криптографическими методами сообщений.

В отношении нарушителя принимается ряд допущений, которые как правило кладутся в основу математических или иных моделей:

1. Нарушитель знает алгоритм шифрования (или выработки ЭЦП) и особенности его реализации в конкретном случае, но не знает секретного ключа.
2. Нарушителю доступны все зашифрованные тексты. Нарушитель может иметь доступ к некоторым исходным текстам, для которых известны соответствующие им зашифрованные тексты.
3. Нарушитель имеет в своем распоряжении вычислительные, людские, временные и иные ресурсы, объем которых оправдана потенциальной ценностью информации, которая будет добыта в результате криптоанализа.

Попытку прочтения или подделки зашифрованного сообщения, вычисления ключа методами криптоанализа называют криптоатакой или атакой на шифр. Удачную криптоатаку называют взломом.

Криптостойкостью называется характеристика шифра, определяющая его стойкость к расшифрованию без знания ключа, т.е. криптоатаке. Показатель криптостойкости - главный параметр любой криптосистемы. В качестве показателя криптостойкости можно выбрать:

– количество всех возможных ключей или вероятность подбора ключа за заданное время с заданными ресурсами;
– количество операций или время с заданными ресурсами, необходимое для взлома шифра с заданной вероятностью;
– стоимость вычисления ключевой информации или исходного текста.

Все эти показатели должны учитывать также уровень возможной криптоатаки. Однако следует понимать, что эффективность защиты информации криптографическими методами зависит не только от криптостойкости шифра, но и от множества других факторов, включая вопросы реализации криптосистем в виде устройств или программ. При анализе криптостойкости шифра необходимо учитывать и человеческий фактор. Например, подкуп конкретного человека, в руках которого сосредоточена необходимая информация, может стоить на несколько порядков дешевле, чем создание суперкомпьютера для взлома шифра.

Современный криптоанализ опирается на такие математические науки как теория вероятностей и математическая статистика, алгебра, теория чисел, теория алгоритмов и ряд других. Все методы криптоанализа в целом укладываются в четыре направления:

1. Статистический криптоанализ. Исследует возможности взлома криптосистем на основе изучения статистических закономерностей исходных и зашифрованных сообщений. Его применение осложнено тем, что в реальных криптосистемах информация перед шифрованием подвергается сжатию, превращая исходный текст в случайную последовательность символов, или в случае гаммирования используются псевдослучайные последовательности большой длины.

2. Алгебраический криптоанализ. Он занимается поиском математически слабых звеньев криптоалгоритмов. К примеру, в 1997 году в эллиптических системах был выявлен класс ключей, которые существенно упрощали криптоанализ.

3. Дифференциальный, или разностный, криптоанализ. Основан на анализе зависимости изменения шифрованного текста от изменения исходного текста. Впервые использован Мерфи, улучшен Бихэмом и Шамиром для атаки на DES.

4. Линейный криптоанализ. Метод, основанный на поиске линейной аппроксимации между исходным и шифрованным текстом. Предложенный Мацуи, также впервые был применен при взломе DES. Как и дифференциальный

анализ в реальных криптосистемах может быть применен только для анализа отдельных блоков криптопреобразований.

Опыт взломов криптосистем, в частности, конкурсов, которые регулярно устраивает RSA DataSecurity, показывает, что главным методом остается "лобовая" атака - проба на ключ. Также, как показывает опыт, криптосистемы больше страдают от небрежности в реализации.

Принято различать несколько уровней криптоатаки в зависимости от объема информации, доступной криптоаналитику. Грубо можно выделить три уровня криптоатаки по нарастанию сложности:

1. Атака по шифрованному тексту (Уровень КА 1). Нарушителю доступны все или некоторые зашифрованные сообщения.

2. Атака по паре "исходный текст - шифрованный текст". (Уровень КА 2). Нарушителю доступны все или некоторые зашифрованные сообщения и соответствующие им исходные сообщения.

3. Атака по выбранной паре "исходный текст - шифрованный текст". (Уровень КА 3). Нарушитель имеет возможность выбирать исходный текст, получать для него шифрованный текст и на основе анализа зависимостей между ними вычислять ключ.

Все современные криптосистемы обладают достаточной стойкостью даже к атакам уровня КА 3, то есть когда нарушителю доступно, по сути, шифрующее устройство.

ОБЗОР ШИРОКО ИСПОЛЬЗУЕМЫХ СОВРЕМЕННЫХ КРИПТОГРАФИЧЕСКИХ АЛГОРИТМОВ

Нет никаких сомнений, что для того чтобы сделать современные криптосистемы такими, какими мы знаем их сейчас - на порядок более стойкими к криптоанализу нежели их предшественники, ученые и исследователи применяли новейшие методики и собственные знания для создания новых криптосистем, обладающих сравнительно большой стойкостью к различным методам криптоаналитиков. Разрабатывались новые пути подхода к решению поставленных задач, постоянно велись поиски наилучшего алгоритма шифрования. Поскольку, данные изыскания большей частью велись обособленно, небольшими группами исследователей, получилось так, что сегодня мы имеем ряд популярных решений, обладающих как своими несомненными преимуществами, так явными, или не очень, недостатками. Некоторые из них, наиболее популярные и востребованные, хотелось бы особо отметить.

Начать следует с описания разбиения криптографических алгоритмов на симметричные и асимметричные.

Как правило, когда говорят о симметричных криптосистемах, подразумевают криптосистемы, использующие один и тот же ключ, как для шифрования, так и для последующей расшифровки передаваемого сообщения.

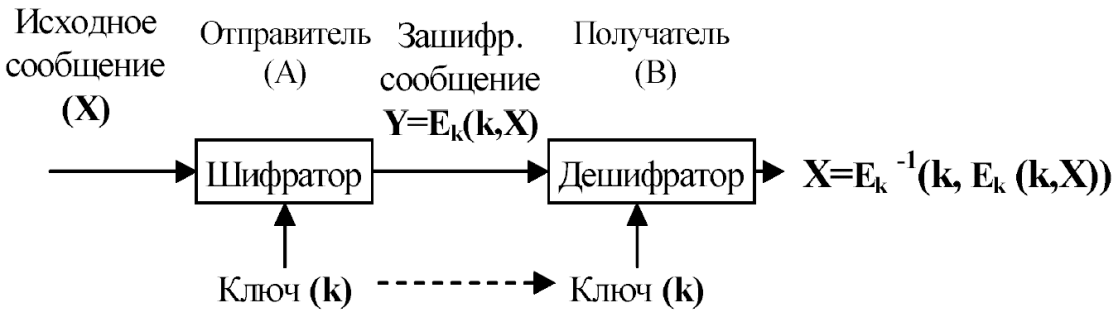


Рисунок 1.1. Схема симметричной криптосистемы.

Данная реализация криптосистемы требует предварительной выработки секретного ключа сторонами, участвующими в передаче шифрованной информации. Особо стоит отметить, недопустимость компрометации данного ключа. Сами симметричные криптосистемы можно условно разделить на 4 базовых класса:

- моно- и многоалфавитные подстановки;
- перестановки;
- блочные шифры;
- гаммирование.

В моноалфавитных подстановках символы исходного текста заменяются на другие из того же алфавита, по некоторому правилу. В свою очередь, многоалфавитные подстановки используют произвольное количество алфавитов и, соответственно, каждый символ исходного сообщения ставится в соответствие символ из одного из них по некоторому принципу.

В перестановках в подлежащем зашифрованию тексте, символы меняются местами по некоторому закону. Такой класс криптосистем не является более надежным и, на сегодняшний день, в чистом виде не используется.

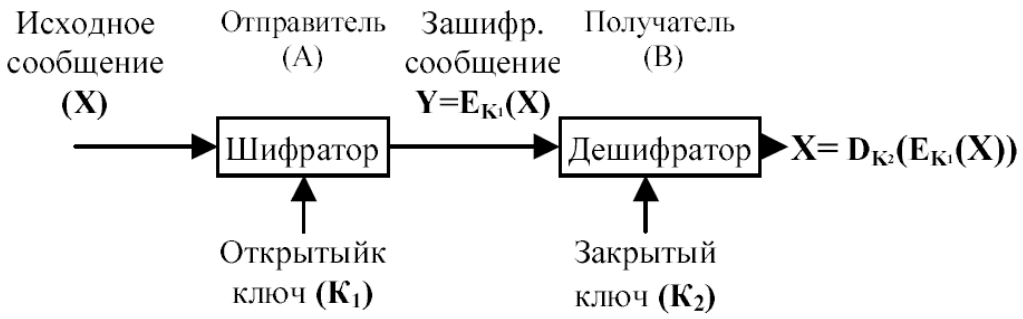
Блочные шифры представляют собой семейство обратимых преобразований. Исходный текст делится на блоки фиксированной длины, после чего производятся некоторые преобразования с этими блоками. Данная разновидность симметричных криптосистем является наиболее распространенными и широко применяемыми на практике.

Гаммирование представляет собой преобразование исходного текста, при котором данные символы складываются с символами псевдослучайной последовательности по модулю, равному мощности алфавита.

Как было сказано ранее, данное разделение является весьма условным и при некоторых условиях, некоторые из них могут рассматриваться как частные случаи других, или применяться неразрывно друг от друга. К примеру, то же гаммирование, при условии использования истинно случайной

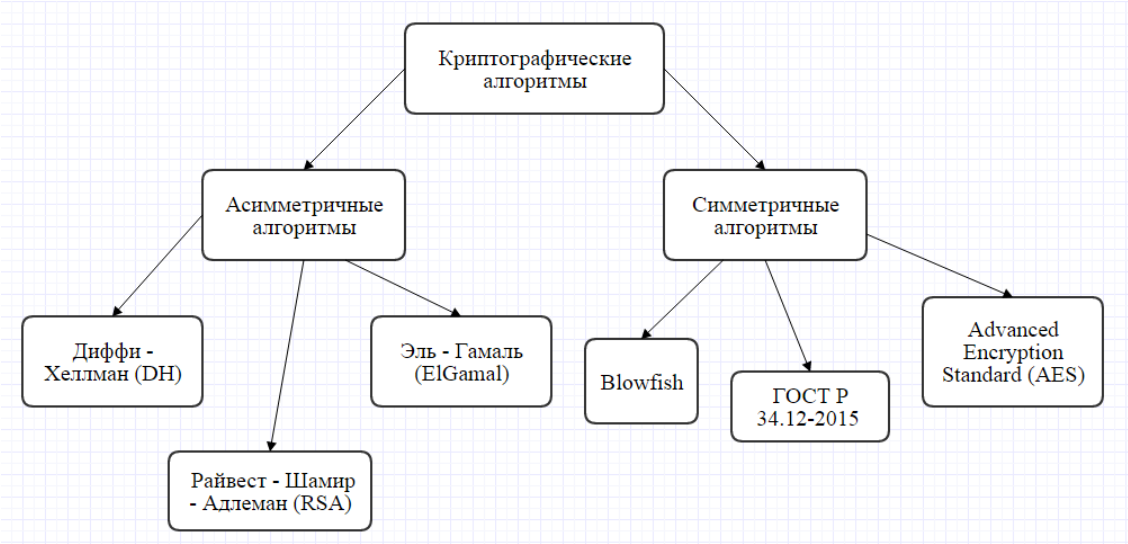
последовательности, как вариант снятой с физического датчика, и применения ее фрагментов только один раз, приводит к тому, что она является криптосистемой с одноразовым ключом. Псевдослучайная последовательность может вырабатывается с помощью блочного шифра.

Асимметричные криптосистемы в своей работе используют пару различных, связанных между собой некоторой зависимостью, ключей, отрытого и секретного. Важным условием является невозможность, или невероятно высокая время затратность, установления одного из ключей, зная при этом другой. Как правило, один из пары этих ключей делается общедоступным, в связи с чем данную криптосистему так же называют криптосистемой с открытым ключом.



Криптосистема с открытым ключом определяется тремя алгоритмами: генерацией ключей, шифрованием и расшифрованием. Алгоритм генерации ключей открыт, и каждый может подать на вход случайную строку r соответствующей длины и получить ключевую пару (k_1, k_2). Один из ключей, к примеру, это может быть k_1, опубликовывается в отрытом доступе. Другой же - сохраняется в тайне. Соответственно, k_1 будем называть открытым, k_2 секретным ключами. Алгоритмы шифрования E_{k_1} и расшифрования D_{k_2} таковы, что для любого открытого текста $X : D_{k_2}(E_{k_1}(X)) = X$.

Далее будут рассмотрены наиболее популярные представители симметричных и асимметричных криптосистем.



Криптосистема Эль – Гамаля (ElGamal)

Система Эль – Гамаля – это криптосистема с открытым ключом, основанная на проблеме дискретного логарифмирования. Система включает как алгоритм шифрования, так и алгоритм цифровой подписи.

Множество параметров системы включает простое число p и целое число g степени которого по модулю p порождают большое число элементов Z_p. У пользователя A есть секретный ключ a и открытый ключ y, где $y = g^a \pmod p$. Предположим, что пользователь B желает послать сообщение m пользователь A. Сначала B выбирает случайное число k, меньшее p. Затем он вычисляет:

$$y_1 = g^k \pmod p$$

$$y_2 = m \oplus (y^k \pmod p)$$

где $\oplus$ обозначает побитовое «исключающее ИЛИ». B посылает A пару (y_1, y_2). После получения шифрованного текста пользователь A вычисляет:

$$m = (y_1^a \bmod p) \oplus y_2$$

Существуют так же вариации данной системы, где вместо «исключающего ИЛИ» используется умножение по модуль простого числа p. Это удобнее в том смысле, что в первом случае текст, или значение хэш – функции, нужно разбивать на блоки длины равной длине $y^k \pmod p$. Во втором случае в этом нет

необходимости, в следствии чего можно обрабатывать блоки фиксированной длины, но меньшей, чем длина числа p. В этом случае уравнение расшифровки выглядит следующим образом:

$$m = \frac{y_2}{y_1^k} \bmod p$$

Данной криптосистеме присуще ряд недостатков таких как отсутствие семантической стойкости и делимость шифра, что в свою очередь ведет к использованию криптосистемы в связке с другими методами, или некоторой модернизации исходного метода. В противном случае, остается возможность проведения некоторых атак, о которых будет сказано выше, и соответственно надежность системы ставится по вопрос.

Криптосистема, основанная на проблеме Диффи – Хеллмана (DH)

Данная система шифрования была представлена Мишелем Абдаллой, Михиром Беллэром и Филиппом Рогэвэем в рамках европейского проекта NESSIE (NewEuropeanSchemesforSignatures, IntegrityandEncryption).

Данная криптосистема реализуема на основе любой циклической группы, для которой может быть сформулирована проблема Диффи – Хеллмана, к примеру, в $\{Z_p^*\}$ или в группе точек на эллиптической кривой. Система строится из криптографических примитивов низкого уровня: групповой операции, симметричного шифра, функции хэширования и алгоритма вычисления кода аутентификации сообщения –имитовставки (MAC). Ее стойкость может быть доказана на основе предположения о сложности решения соответствующей проблемы Диффи – Хеллмана и предположении о стойкости входящих в схему симметричных примитивов.

Дадим описание криптографических примитивов, входящих в данную систему.

Циклическая группа $G = \{g\}$. Используется мультипликативная запись групповой операции. Алгоритмы, реализующие эту операцию, работают с

представлениями элементов группы в виде битовых строк фиксированной длины $gLen \in N$. Способ кодирования $G \to \{0,1\}^{gLen}$ не фиксируется и может выбираться из соображений эффективности.

Код аутентификации сообщения позволяет пользователям, обладающим общим секретным ключом, выработать битовую строку для аутентификации и проверки целостности данных. Пусть $Msg = \{0,1\}^*$ - пространство сообщений, $mKey = \{0,1\}^{mLen}$ - пространство ключей для вычисления MAC для некоторого $mLen \in N$, $Tag = \{0,1\}^{tLen}$ - включающее множество всех возможных значений MAC для некоторого $tLen \in N$. В этих обозначениях код аутентификации сообщений представляет собой пару алгоритмов $MAC = \{MAC.gen, MAC.\mathrm{ver}\}$. Алгоритм генерации MAC определяется как отображение $MAC.gen(k, x) : mKey \times Msg \to Tag$ и может быть вероятностным.

Алгоритм верификации MAC является отображением:

$$MAC.ver(k, x, \tau) : mKey \times Msg \times Tag \to \{0,1\}$$

со свойством $MAC.ver(k, x, MAC.gen(k, x)) = 1$.

В качестве MAC можно использовать, к примеру, блочный шифр с достаточной длиной блока и ключа в режиме сцепления блоков шифрованного текста.

Симметричный шифр позволяет пользователям, обладающим общим секретным ключом, обеспечить секретность. Пусть Msg, как и ранее, пространство сообщений, $eKey = \{0,1\}^{eLen}$ - пространство ключей для некоторого $eLen \in N$, $Ctext = \{0,1\}^*$ - включающее множество всех возможных значений шифрованного текста и $Coins = \{0,1\}^\infty$ - множество строк бесконечной длины. В этих обозначения шифр представляет собой пару алгоритмов $SYM = \{SYM.enc, SYM.dec\}$. Алгоритм зашифрования определяется как отображение:

$$SYM.enc(k, x, r) : eKey \times Msg \times Coins \to Ctext$$

Алгоритм расшифрования является отображением:

$$SYM.\mathrm{dec}(k, y) : eKey \times Ctext \to Msg \cup \{BAD\}$$

где значение BAD выдается, если шифртекст y не является результатом зашифрования никакого открытого текста.

Асимметричный шифр. Пусть Msg, $Ctext$, $Coins$ определены как и ранее, $PK \subseteq \{0,1\}^*$, $SK \subseteq \{0,1\}^*$ - множества открытых и секретных ключей соответственно. Асимметричный шифр определяется как тройка алгоритмов. Алгоритм зашифрования является отображением:

$$ASYM.enc(pk, x, r) : PK \times Msg \times Coins \to Ctext$$

А расшифрования:

$$ASYM.enc(sk, y) : SK \times Ctext \to Msg \cup \{BAD\}$$

Алгоритм выработки ключа в качестве аргумента берет строку $r \in Coins$ и выдает пару ключей $(pk, sk) \in \mathrm{PK} \times \mathrm{SK}$.При этом должно выполняться следующее свойство:

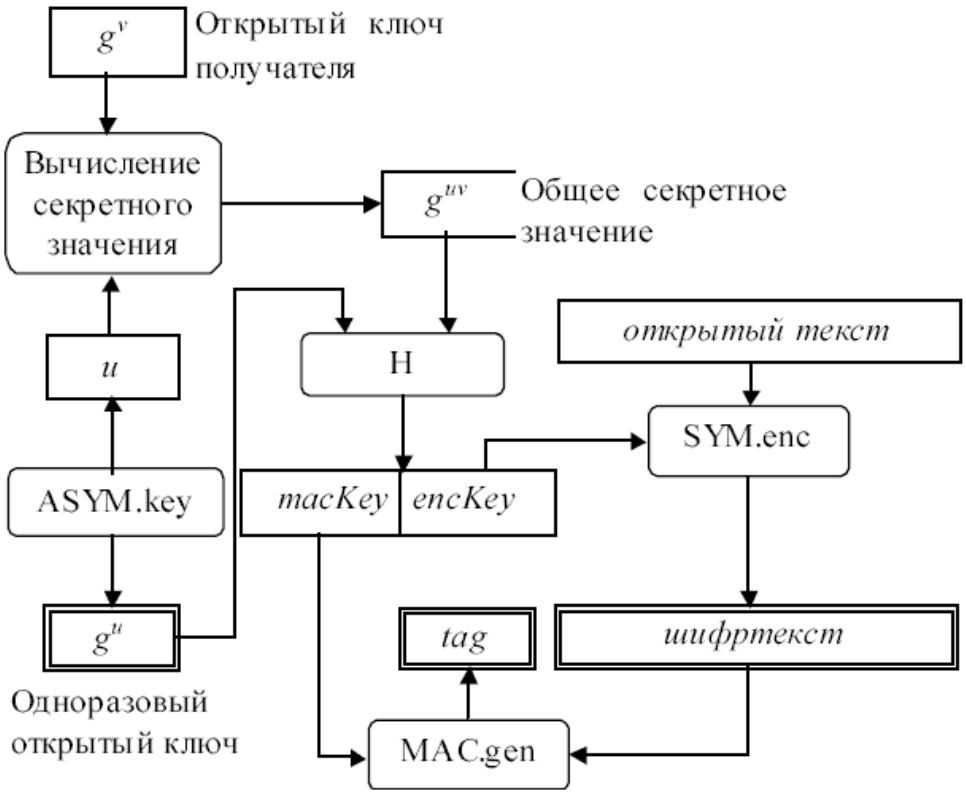
$$\forall (pk, sk) : \exists \mathrm{r}' \in Coins : (pk, sk) = \mathrm{ASYM.key}(\mathrm{r}'), \forall r \in Coins\ \forall x \in Msg\ \ ASYM.dec(sk, ASYM.enc(pk, x, r)) = x$$

Функция хэширования является отображением следующего вида:

$$H : \{0,1\}^{2gLen} \to \{0,1\}^{mLen+eLen}$$

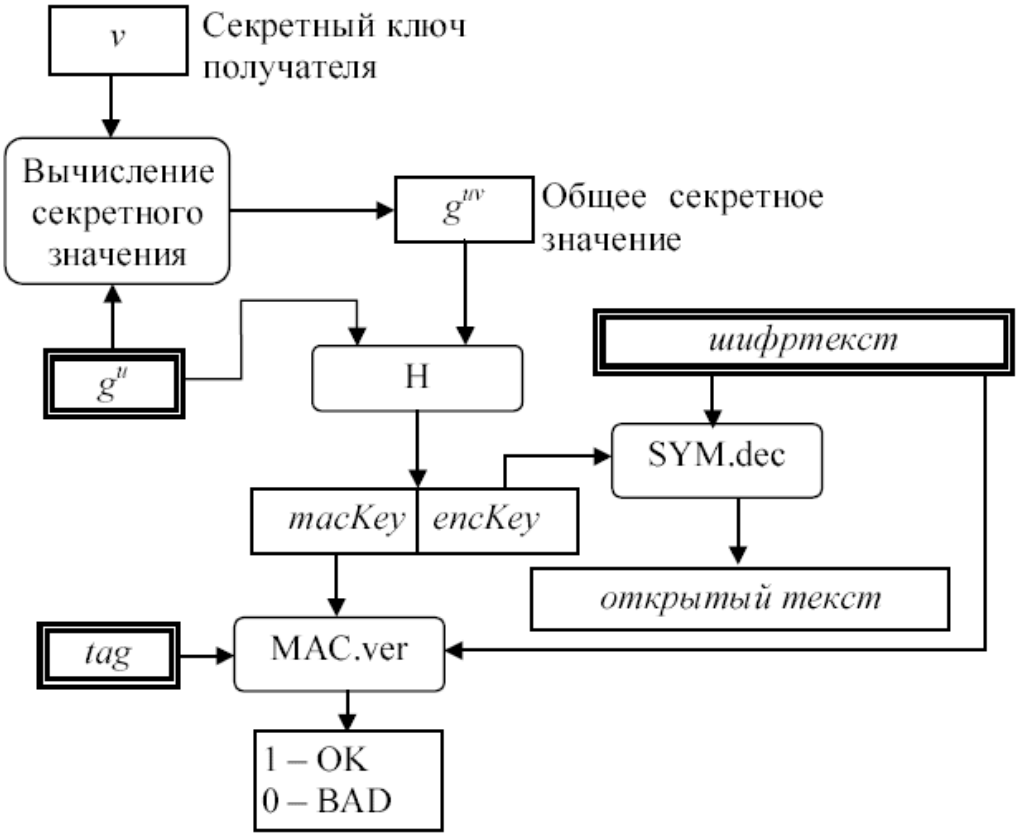
Теперь мы можем описать криптографические примитивы, непосредственно составляющие рассматриваемую криптографическую систему.

Графически процесс зашифрования представлен далее на рисунке.



Все ключевые пары в данном алгоритме выбираются так же, как и в криптосистеме Эль – Гамаля, т.е. пара $(pk, sk) \equiv (g^v, v)$ для некоторого случайного v. При отсутствии сообщения выбирается некоторое случайное значение u и получателю отсылается g^u, что обеспечивает неявный обмен ключами по схеме Диффи – Хеллмана. Таким образом, зашифрованное сообщение состоит из одноразового открытого ключа, текста, зашифрованного симметричным шифром, и кода аутентификации сообщения, выработанного с помощью алгоритма *MAC.gen*.

Процесс расшифрования и аутентификации графически представлен на рисунке ниже. Элементы принятого сообщения также выделены двойной рамкой.



Рассмотренная криптосистема является семантически стойкой и неделимой. В частности, неделимость обеспечивается тем, что значение g^u подается на вход функции хеширования. Ее эффективность по существу та же, что и у Эль – Гамаля, т.е. для зашифрования требуются две операции возведения в степень. А для расшифрования – всего одна. Таким образом. Скорость шифрования для достаточно больших сообщений будет определяться скоростью работы симметричного шифра и алгоритма вычисления кода аутентификации сообщения.

Криптосистема Ривеста – Шамира – Адлемана

Система Ривеста – Шамира – Адлемана, более известная как RSA, аббревиатура, полученная от первых букв создателей, представляет собой криптосистему, стойкость которой основана на сложности решения задачи разложения числа на простые сомножители. Дадим краткое описание алгоритма.

Пользователь A выбирает пару различных простых чисел p_A и q_A, вычисляет $n_A = p_A q_A$ и выбирает число d_A, такое что $НОД(d_A, \varphi(n_A)) = 1$, где $\varphi(n)$ - есть функция Эйлера от n. Если $n = pq$, где p и q - простые числа, то $\varphi(n_A) = (p-1)(q-1)$. Затем он вычисляет величину e_A, при этом должно выполняться:

$$d_A \cdot e_A = 1 (\mathrm{mod}\, \varphi(n_A))$$

затем открытый ключ (e_A, n_A) пользователя A публикуется в открытом доступе.

Теперь пользователь B, который хочет передать сообщение пользователю A представляет исходный текст в виде $x = (x_0, x_1, \ldots, x_{n-1}), x \in Z_n, 0 \le i \le n$, по основанию n_A :

$$N = c_0 + c_1 n_A + \ldots$$

Пользователь B зашифровывает текст при передаче его пользователю A, применяя к коэффициентам c_i отображение $E_{e_A n_A}$:

$$E_{e_A n_A} : c \to c^{e_A} (\mathrm{mod}\, n_A)$$

получая зашифрованное сообщение N'. В силу выбора чисел d_A и e_A, отображение $E_{e_A n_A}$ является взаимно однозначным, и обратным к нему будет отображение:

$$E_{e_A n_A} : c \to c^{d_A} (\mathrm{mod}\, n_A)$$

Пользователь A производит расшифрование полученного сообщения N', применяя $E_{d_A n_A}$.

Для того чтобы найти отображение $E_{d_A n_A}$, обратное по отношению к $E_{e_A n_A}$, требуется знание множителей $n_A = p_A q_A$. Время выполнения наилучших из известных алгоритмов разложения при $n > 10^{145}$ выходит за пределы современных вычислительных возможностей.