ЦИФРОВАЯ КУЛЬТУРА
УНИВЕРСИТЕТ ИТМО

Информационная безопасность

Высшая школа цифровой культуры
Университет ИТМО
dc@itmo.ru

Содержание

1 Введение в информационную безопасность

Основными характеристиками информации с точки зрения ее безопасности принято считать:

- конфиденциальность;
- целостность;
- доступность.

Эти три параметра называют *триадой информационной безопасности* и в английской аббревиатуре они легко запоминаются как сокращение одной очень известной организации – **CIA** (Confidentiality, Integrity, Accessability).

Что же понимается под этими терминами? **Конфиденциальность** информации - это субъективно определяемая характеристика информации, указывающая на необходимость введения ограничений на круг субъектов, имеющих доступ к данной информации. Эта характеристика обеспечивается способностью системы сделать указанную информацию недоступной для субъектов, не имеющих полномочий на доступ к ней.

Целостность информации – свойство информации существовать в неискаженном виде. При этом предполагается, что защитить от искажений информацию в большинстве случаев не представляется возможным и соответственно задача целостности переформулируется как задача обнаружения любых несанкционированных искажений информации.

Доступность информации – свойство системы, в которой хранится, обрабатывается и передается информация, обеспечивать своевременный беспрепятственный доступ субъектов к интересующей их информации и готовность соответствующих автоматизированных служб к обслуживанию поступающих от субъектов запросов всегда, когда в обращении к ним возникает необходимость. При этом подразумевается, что у субъектов есть право на такие действия.

Для того чтобы понимать разнообразие и сложность задач защиты информации в сложных системах необходимо иметь представление о том, каким образом информация в них представляется.

Имеется пять основных уровней представления информации:

- уровень носителей информации;
- уровень средств взаимодействия с носителем информации;
- логический уровень;
- синтаксический уровень;

- семантический уровень.

Рассмотрим теперь более подробно что же собой представляет информация на каждом из этих уровней. По своей природе информация не материальна и соответственно не доступна человеку в чистом виде. Для того, чтобы человек воспринял информацию, должен быть материальный носитель: другой человек, вещество, энергия. Если информация содержится на материальном носителе, то она становится доступной человеку. Таким образом, информация, являясь предметом защиты, требует защищенности тех объектов, в которых она присутствует в той или иной материальной форме.

Информация о носителе может быть разделена на две категории:

- признаковая информация: информация носителя «о себе», о видовых признаках: форма, размер, структура, химические и физические свойства, энергетическая параметры и так далее;

- семантическая информация: информация, не зависящая от вида носителя, продукт абстрактного мышления на языке символов.

Роль человека по отношению к информации многообразна, человек может быть:

- носителем информации;

- генератором новой информации;

- источником информации;

- владельцем информации;

- пользователем информации.

По отношению к информации в смысле ее защиты человек может играть противоположные роли:

- он может быть ее защитником;

- он может быть нарушителем всей триады безопасности.

Как носитель информации человек нуждается не только в физической защите. Его следует защищать от избыточной, бесполезной и ложной информации, от разрушающей информации, представляющей собой аналог деструктивного вируса. Примером такой разрушающей информации может быть, например, информационно-психологическое оружие.

Вещественные носители разнообразны по своим качествам, среди них есть такие, которые используются уже тысячелетиями, есть созданные в последние годы. К наиболее распространённым в настоящее время относятся: бумага, магнитные носители информации, оптические носители информации энергонезависимая память.

Особенности вещественных носителей:

- придают информации свойство статичности (постоянства во времени), в связи с этим обычно используются для хранения информации;
- информация фиксируется прочно, её трудно уничтожить, не повредив носителя;
- со временем вещественные носители разрушаются и стареют, при этом информация гибнет вместе с носителем;
- запись информации связана с изменением физических и химических свойств носителей.

Вещественные носители, как и любой материальный объект, следует защищать от повреждения, преждевременного износа, хищения, утери.

Необходима также защита при копировании информации. Информация может отделяться от вещественного носителя. **Копирование** – это процесс переноса информации на аналогичный или иной носитель без изменения ее количества и качества. Копирование легко обеспечивается при помощи современных технологий. Для документов на бумажном носителе копирование осуществляется при помощи ксерокса, сканера, фотоаппарата. Для магнитных, оптических и энергонезависимых носителей операция копирования предусмотрена стандартным программным обеспечением. В результате копирования одна и та же информация размещается на разных носителях разнесенных территориально и, следовательно, нужна охрана всех носителей во всех местах их хранения и использования.

Энергетические носители - это электромагнитное и акустическое поля. Особенности энергетических носителей:

- используются в основном для передачи информации;
- не стареют;
- бесконтрольно распространяются в пространстве;
- способны к взаимному преобразованию;
- запись информации связана с изменением параметров поля (различные виды модуляции).

Основные способы защиты информации на энергетическом носителе:

- обеспечение помехоустойчивости при выборе кодирования (модуляции);
- обеспечение требуемой энергетики сигнала;
- защита от утечки, в том числе через побочные электромагнитные излучения и наводки (ПЭМИН);
- защита от перехвата в основном канале.

1.1 Уровни взаимодействия с носителем информации

Непосредственное взаимодействие с носителем не всегда возможно. В большинстве случаев оно осуществляется через сложные технические устройства.

Для защиты на этом уровне нужно следить за исправностью устройств обмена информацией, за отсутствием технических средств несанкционированного доступа к информации (так называемых «закладок»), задачей которых является перехват или перенаправление потока считываемой и записываемой информации.

Логический уровень. На логическом уровне информация представима в виде логических дисков, каталогов, файлов, секторов, кластеров. В современных операционных системах уровни информации, разбитой на отдельные байты, кластеры, сектора не видны пользователю, поэтому часто об этом просто забывают.

Однако, для задач информационной безопасности следует помнить, что, например, удаление информации на высоком логическом уровне, то есть на уровне файла, не приводит к удалению информации на нижних уровнях, откуда она может быть легко считана.

Синтаксический уровень. Синтаксический уровень представления информации связан с ее кодированием. Информация записывается и передаётся при помощи символов. Символ – это некоторый знак, которому придаётся определённый смысл.

Линейный набор символов образует алфавит. В процессе кодирования один алфавит может быть преобразован в другой.

В зависимости от целей различаются следующие виды кодирования:

- с целью устранения избыточности – архивирование для передачи и хранения информации без избыточности;
- с целью обнаружения и устранения ошибок - помехоустойчивое кодирование;

- с целью сделать информацию недоступной - криптографическое преобразование.

Семантический уровень. Семантический уровень представления информации связан со смыслом передаваемой информации. Одинаковые лексические конструкции могут иметь различный смысл в разном контексте. Использование профессионализмов, многозначных слов и слов, значение которых изменилось с течением времени, может исказить смысл информации. В информационной безопасности под семантическим уровнем зачастую понимают стеганографическое преобразование информации.

Очевидно, что первые три уровня относятся к организационно-техническим методам защиты информации и затрагивают не саму информацию, а системы, обеспечивающие ее хранение, передачу и обработку, в то время как четвертый и пятый уровни относятся к обработке собственно информации для обеспечения ее защиты. Рассмотрим теперь, какие же методы используются в этом случае для обеспечения конфиденциальности, целостности и доступности информации.

2 Алгоритмы и системы шифрования

Общая классификационная схема методов обработки информации для ее защиты представлена на рисунке. По-прежнему защиту информации можно представить, как триаду, а именно, конфиденциальность, целостность и доступность.

Рис. 1: Методы обработки информации

Наиболее известным и старым способом защиты конфиденциальности информации является ее шифрование. Основными элементами процесса

шифрования является исходный текст, ключи для шифрации и дешифрации, и алгоритмы шифрования и дешифрации.

Принято считать, что алгоритмы шифрования и дешифрации не являются секретными и, значит, известны всем. Очевидно, что в таком случае, наибольшую ценность представляет ключ для дешифрации, который и принято называть секретным ключом. Системы шифрования, в которых ключ шифрования и дешифрации совпадают, либо ключ дешифрации легко (без какой-либо секретной информации) получается из ключа шифрования принято называть симметричными.

Рис. 2: Симметричное шифрование

Несимметричные системы шифрования предполагают наличие некоторой секретной информации (потайной двери – “trap door”) зная которую можно вычислить секретный ключ из ключа шифрования. Ключ шифрования в таких системах называется открытым ключом. Отсюда второе название таких систем шифрования – системы шифрования с открытым ключом.

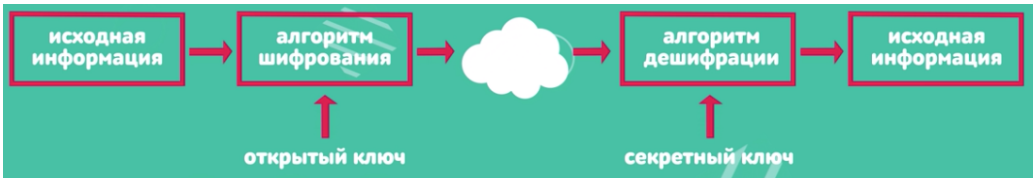

Рис. 3: Несимметричное шифрование

2.1 Симметричные системы шифрования

Рассмотрим теперь более подробно симметричные системы шифрования.

Принято говорить о двух вариантах симметричного шифрования – **блоковом** и **потоковом**.

В первом случае ключ шифрования остается неизменным в процессе шифрации, в то время как во втором он постоянно меняется.

Следует отметить, что если исторически первый блоковый шифр неизвестен и в большинстве случаев про эти алгоритмы шифрования принято говорить, что «человечество начало шифровать свои сообщения практически одновременно с тем как появилась письменность», то дату создания и автора первого потокового шифра принято называть как 1917 год и сотрудник американской компании AT&T Гилберт Вернам.

Потоковые шифры. В 1919 Вернам запатентовал систему автоматического шифрования телеграфных сообщений, в которой каждый символ в сообщении преобразовывался побитовым XOR (исключающее ИЛИ) с ключом записанным на бумажной ленте.

В 1946 году, используя идею шифра Вернама, Клод Шенон определил так называемый «совершенный» шифр, использующий для каждого нового сообщения новый случайный ключ, причем длина такого ключа совпадала с длиной шифруемого сообщения.

Среди блоковых шифров определяют два вида –шифры подстановки (замены) и перестановки.

Шифры перестановки. Простейшим примером шифра перестановки является шифр, использовавшийся еще во времена Древней Спарты. Ключом такого шифра была цилиндрическая палочка - скита́ла или сцита́ла (от греческого слова «жезл»), а шифрование выполнялось следующим образом:

- узкая пергаментная лента наматывалась по спирали на цилиндрическую палочку;
- шифруемый текст писался на пергаментной ленте по длине палочки, после того как длина палочки оказывалась исчерпанной, она поворачивалась и текст писался далее, пока либо не заканчивался текст, либо не исписывалась вся пергаментная лента. В последнем случае использовался очередной кусок пергаментной ленты.

Расшифровка выполнялась с использованием палочки такого же диаметра. Таким образом, длина блока шифруемого текста определялась длиной и диаметром палочки, а само шифрование заключалось в перестановке символов исходного текста в соответствии с длиной окружности палочки. Скитала, как инструмент для шифрования сообщений, был изобретён спартанцами в III веке до н. э.

Конечно, в наше время такими простыми шифрами никто не пользуется, однако шифр перестановки используется как элемент любого современного блокового шифра, в том числе и в российском и в зарубежных стандартах шифрования.

Шифры подстановки. Простейшим вариантом подстановочного шифра является шифр Цезаря , в котором каждый символ открытого текста заменяется символом, находящимся на некотором постоянном числе позиций левее или правее него в алфавите.

Шифр Цезаря называют в честь Гая Юлия Цезаря, который согласно «Жизни двенадцати цезарей» Светония использовал его со сдвигом 3, чтобы защищать военные сообщения для секретной переписки со своими генералами.

Например, для русского алфавита, в шифре со сдвигом вправо на 3, А была бы заменена на Г, Б станет Д, и так далее до последней буквы алфавита Я, которая была бы заменена на В.

2.2 Несимметричная криптография

Несимметричная криптография появилась значительно позже симметричной и первые открытые публикации датируются 1976 годом. История создания криптографических алгоритмов не менее загадочна и секретна чем сами алгоритмы.

Идея передачи секретной информации по незащищенному каналу была первоначально предложена James H. Ellis в 1970 году. Затем Ellis, Cocks и Williamson в 1973 году предложили идею алгоритма RSA, однако результат был засекречен Government Communications Headquarters (Великобритания) и лишь 18 декабря 1997 года, Clifford Cocks анонсировал его, сделав достоянием широкой общественности.

К сожалению James Ellis умер 25 ноября 1997 года за месяц до публичного анонсирования этого факта. В 2010 Malcolm Williamson, Clifford Cocks и James Ellis получили престижную награду Milestone Award Института IEEE (IEEE) за развитие криптографии с открытым ключом.

Рассмотрим теперь, какой же алгоритм был предложен этими тремя британцами.

Алгоритм WCE. В качестве секретного ключа выбираются два большие простые числа p и q. Открытым ключом является их произведение $N = p \cdot q$. Сообщение m должно быть целым положительным числом $m \in \mathbb{Z}$, меньше N.

Для того чтобы зашифровать сообщение его необходимо возвести в степень открытого ключа N и результат взять по модулю N (то есть вычислить остаток от деления результата на число N):

$$e \equiv m^N \pmod{N}.$$

Таким образом, для шифрации достаточно знания только открытого ключа.

При этом, отметим здесь, что открытый ключ в алгоритме WCE представляет собой одно число N, являющееся произведением двух секретных простых чисел p и q таких, что:

$$\text{НОД}(p, (q-1)) = 1,\ \text{НОД}(q, (p-1)) = 1,$$

где НОД – наибольший общий делитель.

Для того чтобы расшифровать сообщение e необходимо знание секретного ключа. Для его вычисления первоначально находится функция Эйлера числа N:

$$\varphi(N) = (p-1) \cdot (q-1),$$

для чего необходимо знать числа p и q секретного ключа.

Затем находится вспомогательное число c:

$$c \equiv N \pmod{\varphi(N)}.$$

Затем вычисляется секретный ключ d:

$$d \cdot c \equiv 1 \pmod{\varphi(N)}.$$

О функции Эйлера и операциях сравнения можно подробней изучить в дополнительных материалах, доступных в разделе текстовых материалов лекции.

Здесь следует заметить, что автоматически выполняется очень важное свойство для корректной работы расширенного алгоритма Евклида. А именно – наибольший общий делитель чисел N и $\varphi(N)$ равен 1.

И, наконец, зашифрованное сообщение e возводится в степень d:

$$m = e^d = (m^c)^d = m^{1 \pmod{\varphi(N)}} = m \pmod{N}.$$

А теперь посмотрим на официальную, наиболее часто встречающуюся, версию проявления несимметричной криптографии. **Протокол Диффи-Хеллмана.** В 1976 году американцы Уитфилд Диффи и Мартин Хеллман опубликовали в журнале IEEE Transaction on Information Theory статью "New directions in cryptography"в которой описали схему шифрования без обмена секретным ключом по открытому каналу.

Основной идеей, лежащей в основе предложенного протокола, являлось использование проблемы дискретного логарифма. То есть, отсутствие эффективного алгоритма вычисления числа x, удовлетворяющего следующему соотношению, при известном простом числе p, и целых числах a и b:

$$a \equiv b^x \pmod{p}.$$

Что же собой представляет этот протокол, опубликованный более 40 лет назад и до сих пор эффективно используемый в огромном числе практических приложений обеспечивающих безопасный канал для любых двух устройств, не имевших до того никакой информации друг о друге и использующих для выработки секретного ключа канал связи свободно прослушиваемый всеми.

Первым шагом протокола является выбор сторонами в открытом обсуждении пары чисел: большого простого числа p и b – первообразного корня из 1 по модулю p. На втором шаге каждый из участников протокола выбирает свое секретное число . Соответственно x и y:

$$1 < x < p - 1,\ 1 < y < p - 1.$$

Для обмена по открытому, прослушиваемому каналу участники вычисляют числа a и c соответственно:

$$a = b^x \pmod{p},\ c = b^x \pmod{p}.$$

После получения этих значений участники протокола могут вычислить общий парный ключ K:

$$K = a^y = c^x = b^{xy} \pmod{p}.$$

3 Алгоритмы шифрования

3.1 Алгоритм шифрования RSA

В 1978 году Дональд Райвест, Ади Шамир и Леонард Адлеман запатентовали и опубликовали свой алгоритм, получивший в дальнейшем название RSA.

В том же номере журнала, известный математик и ученый Мартин Гарднер по согласию с авторами алгоритма, опубликовал математическую задачу, получившую название RSA-129.

В условии задачи он указал два числа n – открытый ключ и e – зашифрованный текст.

Длина числа n составляла 129 десятичных знаков, а число $e = 1007$ знаков. За расшифровку текста предполагалась премия в `100$`!

Шифр удалось взломать через 17 лет – около 600 человек объединились в сеть и усилиями 1600 компьютеров за полгода смогли прочитать в 1995 году фразу:

`«The Magic Words are Squesmish Ossifrage»`
`«Волшебные слова - это брезгливая скопа».`

Мало кто знает, но Скопа – это птица, родственник стервятника.

3.1.1 Основные этапы алгоритма RSA

Выбор секретного ключа и вычисление открытого ключа.

1. Выбираем большие простые числа p и q с близким количеством цифр, после чего вычисляем $N = p \cdot q$.

2. Вычисляем функцию Эйлера от N:

$$\varphi(N) = (p-1) \cdot (q-1).$$

3. Случайным образом выбираем число c, взаимно простое с $\varphi(N)$:

$$\text{НОД}(c, \varphi(N)) = 1.$$

4. С помощь расширенного алгоритма Евклида вычисляем число d, такое, что:

$$cd \equiv 1 \pmod{\varphi)(N}.$$

Число d – **секретный ключ**, так же как и простые числа p и q.

Пара чисел $\{c; N\}$ – **открытый ключ**, который распространяется открыто и доступен всем пользователям системы.

3.1.2 Шифрование сообщений с использованием открытого ключа

Сообщением может быть любое целое положительное число m не превосходящее N. При шифрации используется открытый ключ:

$$e \equiv m^c \pmod{N}.$$

3.1.3 Дешифрование с использование секретного ключа

Возводим число e в степень d и ищем остаток при делении на N.

$$e^d = m^{cd} = m^{1+k\varphi(N)} = m \pmod{N} = m \pmod{N}.$$

Если число N имеет размер 100 цифр, то имеется не менее $4{\cdot}10^42$ простых чисел, которые могут делить число N. Если компьютер выполняет 1 миллион операций в секунду, то ему понадобится примерно 1035 лет для вычисления значения функции Эйлера $\varphi(N)$ от числа N без знания исходных чисел p и q.

В алгоритме RSA, выбранном Гарднером для своего конкурса, в качестве p и q использовались 64 и 65-значные простые числа.

Сейчас для алгоритма RSA используют 150-значные простые числа.

3.2 Алгоритм Эль Гамаля

В 1985 году в журнале IEEE Transactions on Information Theory Тахером Эль-Гамалем был предложен алгоритм шифрования, использующий идеи протокола Диффи-Хеллмана.

В отличии от предыдущих авторов Эль-Гамаль не патентовал свою схему и во многом именно вследствие этого она была использована как основа для национальных стандартов в большинстве стран (Россия, США, в странах Европы и других.).

3.2.1 Основные этапы алгоритма Эль-Гамаля

Выбор секретного ключа и вычисление открытого ключа

1. Выбираем большое простое число p и число q – примитивный корень из единицы по модулю p.

2. Случайным образом выбираем число :

$$1 < c < p - 1.$$

3. Вычисляем число b:

$$b \equiv q^c \pmod{p}.$$

Число c является **секретным ключом.** Тройка чисел $\{p; q; b\}$ – **открытый ключ** алгоритма, который распространяется открыто между пользователями системы.

3.2.2 Шифрование сообщений с использованием открытого ключа

Сообщением может быть любое целое положительное число m не превосходящее числа N. При шифрации используется открытый ключ и случайное число r:

$$1 < r < p - 1$$

Таким образом, зашифрованное сообщение e получается как

$$e \equiv m \cdot b^r \pmod{p}.$$

Кроме того, в алгоритме используется вспомогательное число f, вычисляемое следующим образом:

$$f \equiv q^r \pmod{p}.$$

Зашифрованное сообщение представляется парой чисел:

$$\{e; f\}.$$

3.2.3 Дешифрование с использование секретного ключа

Возводим число f в степень секретного ключа c, берем остаток от деления на число p и получаем число $b^r \pmod{p}$:

$$f^c = q^{rc} = q^{cr} = b^r \pmod{p}.$$

Используя расширенный алгоритм Евклида находим число d мультипликативно обратное к b^r по модулю p.

$$d \cdot b^r \equiv 1 \pmod{p}.$$

Теперь осталось умножить на полученное таким образом число d – первое из пары чисел, представленных в зашифрованном сообщении и мы получим исходное сообщение:

$$e \cdot d = m \cdot b^r d \equiv m \cdot 1 = m \pmod{p}.$$

Подробные алгоритмы RSA и Эль-Гамаля и примеры их применения можно изучить в дополнительных материалах, доступных в разделе текстовых материалов лекции.

4 Электронная подпись

Только что мы посмотрели каким образом можно создать общий секретный ключ, используя для этого открытый прослушиваемый канал связи.

Однако шифрация решает лишь одну из трех основных задач информационной безопасности - обеспечение конфиденциальности хранимой, обрабатываемой и передаваемой информации.

К сожалению, это не позволяет предотвратить незаметное изменение критически важной информации. Очевидно, что злоумышленник, зная открытый ключ, которым было зашифровано исходное сообщение, может легко заменить его на другое. Для того, чтобы этого нельзя было сделать требуется использовать некоторую секретную информацию - секретный ключ.

Когда говорят про электронную подпись, то обязательно упоминается некоторая однонаправленная функция - хэш-функция, позволяющая сообщение любого размера преобразовать в "отпечаток"фиксированной длины (например - 256 бит). Такое преобразование и выполняется с помощью хэш-функции.

В английском языке одним из значений слова "hash"является "путаница". И действительно при выполнении хэш-преобразования исходная информация "запутывается"так, что распутать(восстановить) ее обратно практически не представляется возможным.

Для реализации подписи исходная информация сначала хэшируется, а затем подписывается с помощью секретного ключа. Таким образом, возникает первая уязвимость подписи - так называемая коллизия при вычислении хэш-функции.

Очевидно, если два разных сообщения имеют один и тот же результат хэш-функции, то и подписи у этих сообщений будут одинаковые. В этом случае очевидно, можно просто подставить подпись одного сообщения под другим.

Существуют два вида электронной подписи:

- отрицаемая подпись;
- не отрицаемая подпись.

Отрицаемая подпись подразумевает использование одного и того же секретного ключа и при вычислении и при проверке электронной подписи. В этом случае, очевидно, обе стороны могут подписать сообщение и невозможно будет доказать кто же из них на самом деле подписал документ.

Не отрицаемая подпись использует при вычислении секретный ключ пользователя, а при проверке - его открытый ключ. Такой вариант позволяет говорить о том, что подпись может быть сформирована только одним лицом - обладателем секретного ключа, в то время как поверить подпись может любой, кому известен его открытый ключ.

Как мы с вами уже видели на примере алгоритмов несимметричного шифрования, знание отрытого ключа не дает возможности вычислить соответствующий ему секретный ключ.

Рассмотрим теперь алгоритмы электронной подписи, соответствующие приведенным ранее алгоритмам шифрования RSA и Эль-Гамаля.

4.1 Электронная подпись RSA

Используя введенные ранее обозначения, опишем этапы вычисления электронной подписи и ее проверки, считая, что для сообщения m значение хэш-функции равно h и $h < N$.

Тогда подпись s вычисляется следующим образом с использованием секретного ключа d:

$$s \equiv h^d \pmod{N}.$$

После вычисления электронной подписи хранимой или передаваемой информацией является пара - сообщение и подпись:

$$\{m; s\}.$$

Не трудно заметить, что алгоритм подписи для RSA совпадает с алгоритмом расшифровки и естественным образом требует секретного ключа.

Для того чтобы проверить правильность электронной подписи s, то есть проверить не искажен ли исходный документ m в процессе его хранения, обработки или передачи по каналу связи и действительно ли он подписан конкретным лицом, имеющим открытый ключ $\{c; N\}$

Необходимо выполнить следующие действия:

1. Вычислить хэш функцию от проверяемого сообщения m':

$$h' = \mathsf{Hash}(m').$$

2. Проверить выполняется ли равенство используя открытый ключ $\{c, N\}$:

$$s^c \equiv h' \pmod{N}.$$

Очевидно, что для выполнения этих операций необходим только числа N и c открытого ключа.

4.2 Электронная подпись Эль-Гамаля

Подпись Эль-Гамаля реализуется немного сложнее и главное требует использования случайного числа, что с одной стороны затрудняет ее вычисление, а с другой делает ее более защищенной, так как одно и то же сообщение, подписанное тем же пользователем будет иметь каждый раз разную подпись.

При наличии секретного ключа c и открытого $\{p, q, b\}$ нам понадобится еще случайное целое положительное число r, такое что:

$$1 < r < p - 1$$

И наибольший общий делитель r и $p - 1$ равен 1:

$$\text{НОД}(r, (p - 1)) = 1$$

Электронная подпись s для сообщения m со значением хэш-функци $\mathsf{Hash}(m)$ от подписываемого сообщения равным h, вычисляется из соотношения

$$h \equiv f \cdot c + r \cdot s \pmod{()p - 1},$$

где $f = q^r \pmod{p}$.

Не трудно заметить, что здесь опять необходимо найти число мультипликативно обратное к r по модулю $p - 1$ и это можно сделать только в том случае когда числа r и $p - 1$ взаимно просты, то есть:

$$\text{НОД}(r, (p - 1)) = 1.$$

Подписью в системе Эль Гамаля так же, как и при шифровании, являются два числа $\{s; f\}$.

Для проверки целостности сообщения и принадлежности подписи требуется сообщение m', подпись к нему $\{s; f\}$ и открытый ключ пользователя. Если искажений не было, то есть:

$$m' = m, h' = \mathsf{Hash}(m') = \mathsf{Hash}(m) = h$$

и, если подпись была создана пользователем с открытым ключом $\{p; q; b\}$ и соответствующим ему секретным ключом c, то будет выполнено сравнение

$$f^s \cdot b^f \equiv q^h \pmod{p}.$$

Подробные алгоритмы электронной подписи RSA и Эль-Гамаля и примеры их применения можно изучить в дополнительных материалах, доступных в разделе текстовых материалов лекции.

5 Доступность информации

Рассмотрим теперь вопрос, связанный с обеспечением доступности информации. Очевидно, решение этой задачи сталкивается с очевидным противоречием: а именно, с одной стороны информация должна быть недоступна посторонним пользователям, а с другой стороны легальный пользователь должен легко получить доступ к своей информации. То есть можно сказать, что нам необходимо разработать систему разграничения доступа к информации. Основными компонентами такой системы являются:

- идентификация;
- аутентификация;
- авторизация.

На этапе идентификации от пользователя требуется предъявить свое имя, идентификатор, «log in». При этом доступ к информации будет предоставлен лишь в том случае если такое имя в списке легальных пользователей системы разграничения доступа. Понятно, что наличие этапа идентификации – простейший вариант зашиты информации от посторонних.

Более сложным элементом является Аутентификация, при которой от пользователя требуется доказать, что предъявленное им имя действительно принадлежит ему. В настоящее время рассматривают четыре возможных варианта аутентификации: пароль, ключ, биометрия и поведение. Каждый из этих способов обладает своими преимуществами и недостатками относительно остальных. Наиболее часто используемым и привычным является пароль. Он легко меняется, однако имеет достаточно слабую защиту от атаки методом полного перебора или по словарю. В настоящее время требованием к выбору пароля является наличие в нем всех четырех групп символов – заглавные и строчные буквы, цифры и символы. Кроме того, длина пароля в большинстве случаев должна быть не менее 8 символов.

Ключ удобен в использовании, может быть достаточно сложным, например, 256 случайных бит, его не надо запоминать в отличие от пароля, однако им можно воспользоваться без ведома пользователя и в некоторых случаях он может быть скопирован.

Биометрия представляется наиболее простым способом аутентификации для пользователя так как биометрические компоненты у пользователя всегда «под рукой». Основными недостатками этого метода является их ограниченное число и невозможность поменять в случае компрометации. Кроме того, к сожалению, точность распознавания биометрии не столь высока как ключа и пароля.

И наконец, наиболее современный и возможно наиболее интересный метод аутентификации – по поведению. Одним из наиболее простых и очевидных примеров является динамика вашего почерка, то есть последовательность ускорений с которыми вы пишите текст на бумаге. Однако, можно представить себе и более простой вариант так называемого «клавиатурного» почерка» то есть статистики интервалов между нажатиями клавиш на клавиатуре при наборе текста. Конечно, существуют и более сложные способы – например, походка, поведение при работе в Интернете или с электронной почтой и многое другое.

Аутентификация определяет те опции, которые становятся доступны пользователю после успешного прохождения им идентификации и аутентификации. Например, пользователю может быть разрешено читать, изменять, дополнять, уничтожать информацию или возможно проделывать с нею еще какие-либо действия.

В этой лекции мы рассмотрели наиболее простые и важные элементы информационной безопасности. При этом мы использовали простейшие элементы математики – арифметики. За рамками этой лекции остались современные криптографические алгоритмы использующие результаты теории конечных полей, алгебраической геометрии, теории кодирования и другие разделы современной математики. За рамками лекции остались протоколы, активно используемые в повседневной жизни – аутентификация без разглашения, разделение и сборка секрета, кольцевая подпись и многие другие. Однако надеемся, что даже эта краткая лекция помогла понять вам основные цели, задачи и способы их решения, а также заинтересовала вас тематикой информационной безопасности.

6 Элементы теории чисел

6.1 Основные определения

Приведем несколько базовых определений и обозначений, чтобы в дальнейшем общаться на одном языке. Скорее всего, все эти понятия хорошо известны из школы. Чтобы укоротить некоторые формулировки, множество натуральных чисел будем обозначать $\mathbb{N}$. В это множество входят числа $1, 2, 3, 4, ..., 100, ...$ Множество целых чисел будем обозначать $\mathbb{Z}$. Это множество состоит из чисел $..., -100, ..., -2, -1, 0, 1, 2, ..., 100, ...$

Определение 6.1.1 *Говорят, что число $a \in \mathbb{Z}$ делится на число b, если существует такое число $k \in \mathbb{Z}$, что*

$$a = kb.$$

Обозначают это

$$a \vdots b.$$

При этом число b называется делителем числа a, а число a называется кратным числу b.

Полезно заметить, что число k, возникшее в определении, тоже является делителем a, и, в свою очередь, число a является числом, кратным k.

Пример 6.1.1 *Так как $24 = 6 \cdot 4$, то числа 6 и 4 являются делителями числа 24.*

Особую роль играют числа p, имеющие в качестве делителей только числа ± 1 и $\pm p$.

Определение 6.1.2 *Целые числа p, имеющие в качестве делителей только числа ± 1 и $\pm p$ (так называемые несобственные делители), называются простыми. Остальные целые числа называются составными.*

Замечание 6.1.1 *Обычно, но не всегда, простыми числами называют те простые, которые > 1. Мы тоже будем придерживаться этого определения.*

Замечание 6.1.2 *Число 1 в этом случае не является ни простым, ни составным.*

Итак, несколько простых чисел: $2, 3, 5, 7, 11, ...$

Фундаментальную роль в арифметике играет так называемая основная теорема арифметики.

6.2 Основная теорема арифметики

Теорема 6.2.1 (Основная теорема арифметики) *Любое $a \in \mathbb{N}$, $a \neq 1$ может быть быть записано в виде произведения простых чисел*

$$a = p_1 \cdot p_2 \cdot ... \cdot p_k,$$

причем эта запись единственна с точностью до порядка сомножителей.

Казалось бы, для нас эта теорема настолько естественна, что не требует никаких обоснований. Однако, как и любая теорема, она должна иметь доказательство. Попробуйте его привести. Кроме того, используя этот результат, довольно легко доказать, что множество простых чисел бесконечно (так называемая теорема Евклида). Попробуйте это сделать.

Замечание 6.2.1 *Обратите внимание, что если собрать повторяющиеся простые сомножители в представлении*

$$a = p_1 \cdot p_2 \cdot ... \cdot p_k,$$

то мы получим представление

$$a = p_1^{r_1} \cdot p_2^{r_2} \cdot ... \cdot p_t^{r_t},$$

где все $r_1, ..., r_t > 0$.

Пример 6.2.1 *Ясно, что*

$$240 = 2 \cdot 2 \cdot 2 \cdot 2 \cdot 3 \cdot 5.$$

Кроме того, можно записать

$$240 = 2^4 \cdot 3 \cdot 5.$$

Из основной теоремы арифметики следует, что любое целое число, не равное 1, имеет представление

$$a = \pm p_1^{r_1} \cdot p_2^{r_2} \cdot ... \cdot p_t^{r_t},$$

где все $r_1, ..., r_t > 0$. Допуская, что некоторые показатели простых чисел могут быть нулевыми, заключаем, что любые два целых числа могут быть записаны в виде произведения степеней одних и тех же простых чисел (ну, и соответствующего знака), а именно

$$a = \pm p_1^{r_1} \cdot p_2^{r_2} \cdot ... \cdot p_k^{r_k}, \quad b = \pm p_1^{s_1} \cdot p_2^{s_2} \cdot ... \cdot p_k^{s_k}$$

Пример 6.2.2 *Пусть* $a = 240$, $b = -700$. *Тогда*

$$a = 2^4 \cdot 3 \cdot 5, \quad b = -2^2 \cdot 5^2 \cdot 7$$

и можно записать

$$a = 2^4 \cdot 3 \cdot 5 \cdot 7^0, \quad b = -2^2 \cdot 3^0 \cdot 5^2 \cdot 7.$$

Определение 6.2.1 (НОД) *Пусть даны* 2 *целых числа* a *и* b, *имеющие представления*

$$a = \pm p_1^{r_1} \cdot p_2^{r_2} \cdot ... \cdot p_k^{r_k}, \quad b = \pm p_1^{s_1} \cdot p_2^{s_2} \cdot ... \cdot p_k^{s_k},$$

где $r_1, ..., r_k, s_1, ..., s_k \geq 0$. *Наибольших общим делителем* a *и* b *называется число*

$$p_1^{m_1} \cdot p_2^{m_2} \cdot ... \cdot p_k^{m_k},$$

где $m_k = \min(r_k, s_k)$ *и обозначается* (a, b).

Можно легко показать, что наибольший общий делитель оправдывает свое название – это и правда наибольшее натуральное число, на которое делятся как a, так и b.

Пример 6.2.3 *Возвращаясь к примеру,*

$$a = 2^4 \cdot 3 \cdot 5 \cdot 7^0, \quad b = -2^2 \cdot 3^0 \cdot 5^2 \cdot 7.$$

Тогда

$$(a, b) = 2^2 \cdot 3^0 \cdot 5 \cdot 7^0 = 20.$$

Определение 6.2.2 *Числа, наибольший общий делитель которых равен* 1, *называются взаимно простыми.*

Пример 6.2.4 *Любые два простых числа являются взаимно простыми. Взаимно простыми могут быть и составные числа. Например,*

$$10 = 2 \cdot 3^0 \cdot 5 \cdot 7^0, \quad 21 = 2^0 \cdot 3 \cdot 5^0 \cdot 7.$$

Тогда

$$(10, 21) = 2^0 \cdot 3^0 \cdot 5^0 \cdot 7^0 = 1.$$

6.3 Кольца вычетов

Все мы со школы отлично знаем (и, может быть, даже умеем доказывать) следующую важную теорему о делении с остатком, касающуюся целых чисел $\mathbb{Z}$.

Теорема 6.3.1 *Пусть $a, b \in \mathbb{Z}$. Тогда существуют единственные $p, r \in \mathbb{Z}$, что*

$$a = pb + r,$$

причем $0 \leq r < |b|$

Определение 6.3.1 *Число p часто называют целой частью при делении a на b, а число r – остатком от деления числа a на число b.*

Замечание 6.3.1 *Если остаток равен нулю, то это означает, что число a делится на b.*

Если говорить в терминах обыкновенных дробей, то теорему в вольной форме можно озвучить так: для любой дроби справедливо представление

$$\frac{a}{b} = p + \frac{r}{b},$$

где последняя дробь правильная (то есть модуль числителя меньше модуля знаменателя).
Оказывается, в некоторых задачах бывает удобно отождествлять те числа, которые имеют одинаковый остаток от деления на данное фиксированное число $n \in \mathbb{N}$. Дадим этой фразе более корректную формулировку.

Определение 6.3.2 *Пусть $n \in \mathbb{N}$. Будем говорить, что $a \in \mathbb{Z}$ сравнимо с $b \in \mathbb{Z}$ по модулю n, если $(a - b) \vdots n$. Обозначать мы это будем записью*

$$a \equiv b \pmod{n}.$$

Последнюю запись часто называют сравнением по модулю n.

Конечно, такое определение затуманивает суть происходящего (математики так делать очень любят). По сути, написанное означает, что остатки от деления a и b при делении на число n равны, то есть справедливо следующее наблюдение

Лемма 6.3.1 *$a \equiv b \pmod{n}$ тогда и только тогда, когда остатки от деления чисел a и b на число n равны.*

Пример 6.3.1 *Пусть* $n = 5$, $a = 32$. *Ясно, что*

$$32 = 6 \cdot 5 + 2.$$

Значит, любое число, дающее остаток 2 *при делении на* 5 *будет сравнимо с числом* 32. *Ясно, что общий вид такого числа выписывается явно*

$$b = 5k + 2,\ k \in \mathbb{Z}.$$

Введенное нами определение позволяет отождествлять те числа, которые дают одинаковый остаток от деления на n. Иными словами, все множество целых чисел $\mathbb{Z}$ оказалось разбито на объединение непересекающихся n множеств: $[0]_n, [1]_n, ..., [n-1]_n$, где

$[i]_n$– множество тех целых чисел. что дают остаток i при делении на n.

Обоснование этого следует из следующих важных свойств сравнений, а именно

Лемма 6.3.2 *1.* $a \equiv a \pmod{n}$;

2. $a \equiv b \pmod{n} \Rightarrow b \equiv a \pmod{n}$;

3. $a \equiv b \pmod{n},\ b \equiv c \pmod{n} \Rightarrow a \equiv c \pmod{n}$.

Итак, рассматривая число $n \in \mathbb{N}$, у нас естественным образом возникает n множеств. Дадим формальное определение

Определение 6.3.3 *Множество* $[i]_n$ *(то есть множество чисел, которые при делении на* n *дают в остатке* i*), в которое входит число* $a \in \mathbb{Z}$, *называется вычетом числа* a *по модулю* n.

Определение 6.3.4 *Набор множеств* $[0]_n, [1]_n, ..., [n-1]_n$ *называют полной системой вычетов по модулю* n

Последнее определение обусловлено тем, что по модулю n у целого числа нет других остатков, кроме как $0, 1, 2, ..., (n-1)$. Оказывается, что сравнения обладают следующими важными свойствами, а именно

Лемма 6.3.3 *1. Если* $a \equiv b \pmod{n}$ *и* $c \equiv d \pmod{n}$, *то* $(a+c) \equiv (b+d) \pmod{n}$;

2. Если $a \equiv b \pmod{n}$ *и* $c \equiv d \pmod{n}$, *то* $ac \equiv bd \pmod{n}$;

3. Если $ka \equiv kb \pmod{n}$ *и* $(k, n) = d$, *то* $a \equiv b \pmod{\frac{n}{d}}$;

4. *Если* $a \equiv b \pmod{n}$, *то* $ka \equiv kb \pmod{n}$ *или даже* $ka \equiv kb \pmod{(kn)}$.

Доказательство. Чтобы не быть совсем голословными, давайте докажем, например, свойство 2, которым будем активно пользоваться в дальнейшем. $a \equiv b \pmod{n}$, означает, что $(a-b) \vdots n$, то есть существует число $t \in \mathbb{Z}$, что $a - b = nt$. Аналогично, существует $k \in \mathbb{Z}$, что $c - d = nk$. Тогда

$$ac = (b + nt)(d + nk) = bd + n(bk + td + nkt),$$

где $(bk + td + nkt)$ – целое, что и доказывает утверждение. □

Приведенные свойства позволяют легко вычислять остатки от деления, например, степеней некоторого числа.

Пример 6.3.2 *Пусть требуется найти остаток от деления числа* 13^{11} *на* 35, *то есть найти* $0 \leq x < 35$, *что*

$$x \equiv 13^{11} \pmod{35}.$$

Ясно, что $13 \equiv 13 \pmod{35}$. *Далее,*

$$13^2 = 169 = 140 + 29 = 35 \cdot 4 + 29 \equiv 29 \equiv -6 \pmod{35},$$

$$13^3 = 13^2 \cdot 13 \equiv -6 \cdot 13 \pmod{35},$$

а $-78 \equiv -8 \pmod{35}$, *откуда* $13^3 \equiv 27 \pmod{35}$. *Так как*

$$13^4 = 13^2 \cdot 13^2 \equiv 36 \pmod{35},\ 36 \equiv 1 \pmod{35} \Rightarrow 13^4 \equiv 1 \pmod{35},$$

то

$$13^{11} = 13^8 \cdot 13^3 \equiv 1 \cdot 27 \pmod{35}$$

и

$$13^{11} \equiv 27 \pmod{35}.$$

Пример 6.3.3 *Найти остаток от деления* 2^{100} *на* 125. *Для этого нужно найти такое число* $0 \leq x < 125$, *что* $x \equiv 2^{100} \pmod{125}$.
Так как $2^7 = 128 \equiv 3 \pmod{125}$, *то* $2^{35} = (2^7)^5 \equiv 3^5 \equiv 243 \equiv -7 \pmod{125}$, $2^{50} = 2^{35} \cdot 2 \cdot (2^7)^2 \equiv -126 \equiv -1 \pmod{125}$, *откуда*

$$2^{100} \equiv 1 \pmod{125}$$

6.4 Функция Эйлера и некоторые ее свойства

В теории чисел и приложениях богатое применение имеет так называемая функция Эйлера.

Определение 6.4.1 (Функция Эйлера) *Функция $\varphi : \mathbb{N} \to \mathbb{N}$, возвращающая по числу n количество взаимно простых чисел с ним, не превосходящих n, называется функцией Эйлера. При этом полагают $\varphi(1) = 1$.*

Пример 6.4.1 *Вычислим $\varphi(15)$. Так как $15 = 3 \cdot 5$, то из ряда $1, 2, 3, ..., 15$ нужно исключить все числа, делящиеся либо на 3, либо на 5. Тогда останутся числа*

$$1, 2, 4, 6, 7, 8, 11, 13,$$

то есть $\varphi(15) = 8$.

Из только что данного определения сразу следует, что если p – простое число, то

$$\varphi(p) = p - 1.$$

Действительно, до числа p находится ровно $(p-1)$ натуральное число. Так как p простое, то оно взаимно просто с каждым из них.
Для вычисления функции Эйлера от составного числа можно воспользоваться ее важным свойством, свойством мультипликативности, а именно

Лемма 6.4.1 (О мультипликативности функции Эйлера) *Если $m = nk$, где $m, n, k \in \mathbb{N}$ и $(n, k) = 1$, то*

$$\varphi(m) = \varphi(nk) = \varphi(n)\varphi(k).$$

Если p – простое, $n \in \mathbb{N}$, то

$$\varphi(p^n) = p^n - p^{n-1}.$$

Пример 6.4.2 *Вычислить $\varphi(117)$. Так как $117 = 13 \cdot 3^2$, а $\varphi(13) = 12$, $\varphi(3^2) = 3^2 - 3 = 6$, то*

$$\varphi(117) = \varphi(13)\varphi(3^2) = 12 \cdot 6 = 72.$$

С функцией Эйлера связано еще одно важное утверждение теории чисел, так называемая теорема Эйлера.

Теорема 6.4.1 (Теорема Эйлера) *Пусть $(a, n) = 1$, тогда*

$$a^{\varphi(n)} \equiv 1 \pmod{n}.$$

Вернемся к уже ранее разобранному примеру

Пример 6.4.3 *Найти остаток от деления числа 2^{100} на 125. Так как $125 = 5^3$, то $\varphi(125) = \varphi(5^3) = 5^3 - 5^2 = 100$, откуда*

$$2^{100} \equiv 1 \pmod{125}.$$

6.5 Решение линейных сравнений

Рассмотрим общую задачу о нахождении такого множества X, что для каждого $x \in X$

$$ax \equiv b \pmod{n}.$$

Возможны следующие варианты.
1. $(a, n) = 1$. Мы знаем, что согласно теореме Эйлера,

$$1 \equiv a^{\varphi(n)} \pmod{n}.$$

Тогда

$$ax \equiv ba^{\varphi(n)} \pmod{n}$$

и

$$x \equiv ba^{\varphi(n)-1} \pmod{n}.$$

Если x пробегает полную систему вычетов по модулю n, то ax тоже пробегает полную систему вычетов, а значит только при одном значении x, взятом из полной системы, решение возможно. Тем самым, решение единственно.
2. Если $(a, n) = d > 1$ и $b \vdots d$, то поделим наше сравнение на d и перейдем к сравнению

$$a_1 x \equiv b_1 \pmod{n_1},$$

причем $(a_1, d_1) = 1$. По предыдущему пункту, такое сравнение имеет решение $x \equiv b_1 a_1^{\varphi(n_1)-1} \pmod{n_1}$. Ясно, что исходное сравнение имеет d решений, причем все они могут быть выписаны, как

$$x \equiv b_1 a_1^{\varphi(n_1)-1} + n_1 k \pmod{n}, \quad k = 0, ..., (d-1).$$

3. Если же $(a, n) = d > 1$, но b не делится на d, то сравнение, очевидно, решений не имеет.

Пример 6.5.1 *Решить сравнение*

$$3x \equiv 7 \pmod{5}.$$

Так как $(3, 5) = 1$, *то сравнение имеет единственное решение. Так как* $\varphi(5) = 4$, *то*

$$x \equiv 7 \cdot 3^3 \pmod{5} \text{ } \textit{или} \text{ } x \equiv 189 \pmod{5}.$$

Упростив, можно записать

$$x \equiv 4 \pmod{5}.$$

Замечание 6.5.1 *Часто число* $b \cdot a^{\varphi(n)-1}$ *оказывается очень большим, особенно, если* n *большое. Чтобы уменьшить это число можно поступить либо так, как мы делали в первой части, либо сразу использовать несколько другой подход. Так как*

$$3x \equiv 7 \pmod 5,$$

то

$$3x \equiv 7 + 5 \pmod 5, \text{ откуда } 3x \equiv 12 \pmod 5.$$

Так как $(3, 5) = 1$, *то получим эквивалентное сравнение*

$$x \equiv 4 \pmod 5.$$

Пример 6.5.2 *Решить сравнение*

$$6x \equiv 15 \pmod 9.$$

Так как $(6, 9) = 3$, *то, поделив на* 3, *получим сравнение*

$$2x \equiv 5 \pmod 3.$$

Так как в то же время $2x \equiv 5 + 3 \pmod 3$, *то есть* $2x \equiv 8 \pmod 3$, *получим*

$$x \equiv 4 \pmod 3.$$

Тогда для исходного сравнения получим три решения

$$x \equiv 4 \pmod 9, \quad x \equiv 7 \pmod 9, \quad x \equiv 10 \pmod 9.$$

Последнее можно так же переписать, $x \equiv 1 \pmod 9$.

Пример 6.5.3 *Решить сравнение*

$$2x \equiv 5 \pmod 8.$$

Так как $(2, 8) = 2$, *но* 5 *не делится на* 2, *то решений сравнение не имеет.*

7 Шифрование информации

7.1 Алгоритм шифрования RSA

Рассмотрим подробно алгоритм шифрования RSA – криптографический алгоритм с открытым ключом, который основан на вычислительной сложности разложения на множители больших чисел. Алгоритм генерации ключей может быть записан по шагам следующим образом:

1. Выбирается два различных простых числа p, q.
2. Вычисляется их произведение $N = p \cdot q$, называемое модулем.
3. Вычисляется $\varphi(N) = (p-1) \cdot (q-1)$.
4. Выбирается целое число $1 < c < \varphi(N)$ такое, что НОД$(c, \varphi(N)) = 1$. При этом число c называют открытой экспонентой (public exponent).
5. Вычисляется число d, решая сравнения $cd \equiv 1 \pmod{\varphi(N)}$. При этом число d называется секретной экспонентой.
6. Пара $\{c, N\}$ является открытым ключом, а пара $\{d, N\}$ – закрытым, его держат в секрете.

Предположим, что мы хотим передать сообщение от Боба Алисе. Тогда Боб должен знать открытый ключ Алисы, чтобы зашифровать сообщение, а Алиса должна использовать свой секретный ключ, чтобы его расшифровать. Пусть Боб хочет послать Алисе сообщение m. Сообщением выступают целые числа в диапазоне $0, ..., (N-1)$. Тогда Боб выполняет следующие действия:

1. По открытому ключу $\{c, N\}$ Алисы вычисляет $e = m^c \pmod{N}$.
2. Передает зашифрованное число e.

Чтобы Алиса могла расшифровать сообщение, она поступает по следующему алгоритму:

1. Принимает зашифрованное сообщение e.
2. С помощью своего секретного ключа $\{d, N\}$ вычисляет $x \equiv e^d \pmod{N}$. Число $x = m$ и есть то сообщение m, что зашифровал Боб.

7.1.1 Пример шифрования чисел

Рассмотрим простой пример, демонстрирующий все этапы вычислений. Для начала, сформируем открытый и закрытый ключи, для этого:

1. Выберем, например, простые числа $p = 3$, $q = 7$.
2. Вычислим модуль $N = 3 \cdot 7 = 21$.
3. Найдем функцию Эйлера от модуля: $\varphi(N) = (3-1) \cdot (7-1) = 12$.
4. Выберем число c, меньшее 21 такое, что НОД$(c, 12) = 1$. Например, пусть $c = 5$.
5. Тогда d найдем из сравнения

$$5d \equiv 1 \pmod{12},$$

откуда $5d \equiv 1 + 24 \pmod{12}$ и $d \equiv 5 \pmod{12}$. Ясно, что в качестве d подойдет значение 5, однако, чтобы не путать величины d и c, выберем $d = 17$.

6. Тем самым, пара $\{5, 21\}$ – открытый ключ, а пара $\{17, 21\}$ – секретный.

Пусть Боб хочет передать Алисе сообщение $m = 19$. Тогда он, по открытому ключу $\{5, 21\}$, вычисляет

$$e \equiv 19^5 \pmod{21}.$$

Так как $19^2 = 361 \equiv 4 \pmod{21}$, то

$$19^5 = 19^2 \cdot 19^2 \cdot 19 \equiv 4 \cdot 4 \cdot 19 \pmod{21}.$$

Но $304 \equiv 10 \pmod{21}$, а значит $e = 10$.

Алиса, получив сообщение e, использует свой секретный ключ. Для этого она вычисляет

$$x \equiv 10^{17} \pmod{21}.$$

Так как $10^2 = 100 \equiv 16 \pmod{21}$, $10^4 \equiv 256 \equiv 4 \pmod{21}$, то

$$10^{17} \equiv 4 \cdot 4 \cdot 4 \cdot 4 \cdot 10 = 2560 \equiv 19 \pmod{21}.$$

Тем самым, Алиса получает сообщение 19.

7.1.2 Пример шифрования текстовой информации

Что делать в ситуации, когда нам требуется зашифровать и передать некоторый текст, а не число? Например, наша цель – закодировать фразу

`Сизифов труд`

Каким образом можно поступить? Давайте каждой букве, каждому символу, присвоим некоторое число. В прочем, это давно придумано и называется кодом ASCII. В нем 255 символов, а значит, для успешного кодирования любого символа, модуль N должен быть больше, чем 255. Выберем, для примера, $p = 17$, $q = 23$. Тогда $N = 17 \cdot 23 = 391$. Ясно, что $\varphi(N) = 16 \cdot 22 = 352$. Пусть $c = 301$, тогда d находится из сравнения

$$301d \equiv 1 \pmod{352}$$

и $d = 69$. Тем самым, пара $\{301, 391\}$ – открытый ключ, а пара $\{69, 391\}$ – секретный ключ.

Приступим к кодированию.
Буква `С` в ASCII таблице имеет код 209, тогда ее закодированное значение получим из сравнения

$$e \equiv 209^{301} \pmod{391},$$

откуда $e = 292$.
Следующий символ – буква `и` имеет в ASCII код 232. Закодировав, получим

$$e \equiv 232^{301} \pmod{391},$$

откуда $e = 177$.
Итого, вместо первых двух наших символов мы будем передавать 292177.
Отметим один важный момент. Для того, чтобы процесс раскодирования не вызывал больших трудностей, мы будем кодировать каждый символ тремя цифрами. Например, если закодированное сообщение имеет вид $e = 3$, то мы будем передавать 003, а если сообщение имеет вид $e = 32$, то будем передавать 032. Продолжая процесс, получим полное закодированное сообщение:

$$292177300177316255020121174304156329.$$

Опишем теперь процесс раскодирования.
Делим полученное сообщение на блоки по три цифры

$$292\ 177\ 300\ 177\ 316\ 255\ 020\ 121\ 174\ 304\ 156\ 329,$$

и для раскодирования первой буквы вычисляем, используя секретный ключ $\{69, 391\}$,

$$x \equiv 292^{69} \pmod{391},$$

откуда $x = 209$. Используя таблицу ASCII, числу 209 соответствует буква С. Действуя аналогично, раскодируем фразу обратно.

Отметим несколько отрицательных моментов такого кодирования. Легко видеть, что указанный метод довольно слаб, так как шифрует по буквам, в результате чего одна и та же буква или символ шифруется одним и тем же числом. Таким образом пробелы (если это текст) сразу выдадут свои позиции. Далее можно подобрать коды символов на основе частоты использования букв, например «а» «о», а значит злоумышленник, перехватив зашифрованное сообщение, может расшифровать его и без секретного ключа, и даже не пытаясь его подобрать. Метод можно интуитивно усложнить, если при шифровании каждой буквы на нее будет влиять результат, полученный от предыдущей, например, так:

$$b \equiv (b + a) \pmod{n},$$

где b – текущее значение, а a – предыдущее. Первая буква остается исходной, однако и это легко решается добавлением в начале сообщения произвольной последовательности букв (которые получатель после расшифровки просто проигнорирует). Эти действия заметно усложняют процесс угадывания, и в таком виде алгоритм уже часто применяется в реальной практике.

7.2 Алгоритм шифрования Эль-Гамаля

Алгоритм Эль-Гамаля в чем-то схож с рассмотренным ранее RSA, но, в отличие от последнего, не был запатентован, и потому стал более дешевой альтернативой: не требовалась оплата взносов за лицензию. Алгоритм генерации ключей может быть записан по шагам следующим образом:

1. Выбирается простое число p.

2. Выбирается произвольное целое число q, являющееся так называемым примитивным (чаще – первообразным) корнем по модулю p:

$$q^{\varphi(p)} \equiv 1 \pmod{p},$$

$$q^l \not\equiv 1 \pmod{p} \text{ при } 1 \leq l < \varphi(p).$$

3. Выбирается случайное целое число c такое, что $1 < c < p - 1$.

4. Вычисляется число b из следующего сравнения

$$b \equiv q^c \pmod{p}$$

5. Открытым ключом является тройка $\{p, q, b\}$, закрытым (секретным) ключом — число c.

Нахождение первообразного корня по модулю p сопряжено с некоторыми трудностями, так как проверка условий

$$q^l \not\equiv 1 \pmod{p} \text{ при } 1 \leq l < \varphi(p).$$

требует временных затрат. Для упрощения вычислений часто бывает полезна следующая теорема.

Теорема 7.2.1 *Пусть $\varphi(p)$ раскладывается на простые множители, как*

$$\varphi(p) = p_1^{k_1} \cdot p_2^{k_2} \cdot ... \cdot p_n^{k_n},$$

причем НОД$(g, p) = 1$. Тогда g – первообразный корень по модулю p тогда и только тогда, когда сравнения

$$g^{\frac{\varphi(p)}{p_1}} \equiv 1 \pmod{p},\ g^{\frac{\varphi(p)}{p_2}} \equiv 1 \pmod{p},\ ...\ ,\ g^{\frac{\varphi(p)}{p_n}} \equiv 1 \pmod{p}$$

не имеют решений.

Перейдем непосредственно к шифрованию. Предположим, что мы хотим передать сообщение от Боба Алисе. Тогда Боб должен знать открытый ключ Алисы, чтобы зашифровать сообщение, а Алиса должна использовать свой секретный ключ, чтобы его расшифровать.

Пусть Боб хочет послать Алисе сообщение m не превосходящее числа $N - 1$, где $N = p$. Тогда Боб выполняет следующие действия:

1. Выбирает случайное целое число r такое, что $1 < r < p - 1$.

2. Вычисляются два числа

$$e \equiv m \cdot b^r \pmod{p},$$

$$f \equiv q^r \pmod{p}.$$

3. Пара чисел $\{f, e\}$ является шифротекстом.

Зная закрытый ключ c, Алиса может расшифровать исходное сообщение, получив шифротекст $\{f, e\}$, по следующей готовой формуле:

$$m = f \cdot e^{-c} \pmod{p}.$$

7.2.1 Пример шифрования

Рассмотрим простой пример, демонстрирующий все этапы вычислений. Для начала сформируем ключи, для этого:

1. Выбираем число $p = 17$.

2. Вычисляем первообразный корень q по модулю 17. Можно проверить, что любое из чисел $3, 5, 6, 7, 10, 11, 12, 14$ является первообразным корнем по модулю 17. Мы выберем, например, число $q = 11$.

3. Выбираем случайное число $c = 14 < p - 1$.

4. Находим число b из сравнения

$$b \equiv 11^{14} \pmod{17} \Rightarrow b = 9$$

5. Открытым ключом является тройка $\{17, 11, 9\}$, а закрытым ключом число 14.

Пусть Боб хочет передать Алисе сообщение $m = 13$. Тогда он выбирает случайное число $r = 7 < p - 1$ и вычисляет числа:

$$e \equiv 13 \cdot 9^7 \pmod{17} \Rightarrow e = 9,$$

$$f \equiv 11^7 \pmod{17} \Rightarrow f = 3.$$

Полученный шифротекст $\{3, 9\}$ передается Алисе, которая использует ключ $c = 14$ и дешифрует полученное сообщение.

$$m = 9 \cdot 3^{-14} \pmod{17} \Rightarrow p = 13$$

Тем самым, Алиса получает зашифрованное Бобом сообщение 13.

8 Электронные подписи

8.1 Электронная подпись RSA

Рассмотрим применение алгоритма шифрования RSA для электронной подписи документов. Цифровая подпись необходима в тех ситуациях, когда важно установить изменение данных и подлинность подписавшей стороны. Получатель подписанного документа может использовать цифровую подпись для доказательства третьей стороне того, что подпись действительно сделана отправляющей стороной, а в документ не было внесено изменений.

В ходе подписания документа подсчитывается хеш-функция, которая сократит любой его объем до определенного количества байтов. Хеш-функции бывают различных типов и также влияют на уровень надежности электронной подписи. Таблица ASCII символов в определенном смысле также является хэшем, но на практике текст сначала кодируется, а потом хешируется.

Электронная подпись, в отличие от ассиметричного шифрования, основана на закрытом ключе. Проверить же подпись может любой желающий, используя открытый ключ. Генерация ключей происходит точно так же, как и для шифрования.

Алгоритм подписи может быть записан по шагам следующим образом:

1. Генерация ключей $\{c, N\}$ и $\{d, N\}$ происходит так же, как и в алгоритме RSA, только теперь $\{c, N\}$ – секретный ключ, а $\{d, N\}$ – открытый ключ.

2. Вычисляется хеш-значение h документа или сообщения m.

3. Вычисляется подпись s сообщения m из сравнения

$$s \equiv h^c \pmod{N}.$$

4. Пара $\{m, s\}$ из сообщения и подписи передается адресату.

Принимающая сторона должна иметь возможность проверить, получила ли она истинный документ без изменений, или его подделку. Алгоритм проверки электронной подписи может быть представлен следующими шагами:

1. Проверяющая сторона получает пару из сообщения и подписи $\{m, s\}$.

2. Вычисляется хеш-значение h сообщения m.

3. Вычисляется значение h', используя открытый ключ $\{d, N\}$, из следующего сравнения

$$h' \equiv s^d \pmod{N}$$

4. Если равенство $h = h'$ выполняется, значит подпись верна.

8.1.1 Пример получения электронной подписи

Рассмотрим простой пример, демонстрирующий все этапы вычисления электронной подписи и ее проверки.

Для начала, сформируем открытый и секретный ключи, для этого обратимся к примеру шифрования алгоритмом RSA и возьмем ранее найденные значения. Для $p = 3$, $q = 7$ были найдены: $\{5, 21\}$ – открытый ключ и $\{17, 21\}$ – закрытый. В алгоритме электронной подписи все наоборот: $\{5, 21\}$ – секретный ключ, а $\{17, 21\}$ – открытый.

Пусть Боб хочет передать Алисе сообщение $m = 19$, подписанное секретным ключом $\{5, 21\}$. Тогда он хеширует свое сообщение неким алгоритмом и получает, например, значение $h = 19$, а затем по ключу вычисляет подпись s из сравнения

$$s \equiv 19^5 \pmod{21}.$$

Таким образом $s = 10$.

Алиса получает сообщение m, подпись s, и использует открытый ключ для проверки электронной подписи. Она хеширует полученное сообщение таким же алгоритмом и получает $h = 19$ и вычисляет h' из сравнения

$$h' \equiv 10^{17} \pmod{21}.$$

Таким образом $h' = 19$ и совпадает с исходным хешем h. Это означает что сообщению стоит доверять, и оно действительно отправлено Бобом, а Алиса получила его без каких-либо повреждений, или оно не было подменено во время передачи.

8.2 Электронная подпись Эль-Гамаля

Алгоритм подписи Эль-Гамаля может быть записан по шагам следующим образом:

1. Генерируются открытый ключ $\{p, q, b\}$ и секретный ключ c, согласно алгоритму генерации ключей Эль-Гамаля.

2. Вычисляется хеш-значение h документа или сообщения m.

3. Выбирается случайное целое число r такое, что $1 < r < p - 1$.

4. Вычисляется значение $f = q^r \pmod{p}$.

5. Вычисляется подпись s из сравнения

$$h \equiv f \cdot c + r \cdot s \pmod{p - 1},$$

или

$$s \equiv (h - f \cdot c)r^{-1} \pmod{p - 1}$$

6. Пара $\{s, f\}$ является подписью сообщения или документа.

Зная открытый ключ $\{p, q, b\}$, подпись $\{s, f\}$ сообщения m проверяется следующим образом:

1. Проверяются условия $0 < s < p$ и $0 < f < p - 1$.

2. Вычисляется хеш-значение h сообщения m.

3. Подпись считается верной, если выполняется следующее сравнение:

$$b^f \cdot f^s \equiv q^h \pmod{p}$$

8.2.1 Пример получения электронной подписи

Рассмотрим пример. Возьмем уже найденный открытый ключ $\{17, 9, 4\}$ и закрытый ключ 14 из примера на шифрование.

Пусть Боб хочет передать Алисе сообщение $m = 19$, тогда он хеширует свое сообщение неким алгоритмом и получает, например, значение $h = 19$. Далее Боб выбирает случайное число $r = 7 < p - 1$ и вычисляет $f = 9^7 \pmod{17} = 2$. Тогда подпись вычисляется из сравнения

$$s \equiv (19 - 2 \cdot 14) \cdot 7^{-1} \pmod{16} \Rightarrow s = 1.$$

Алиса, получает сообщение m, подпись $\{s, f\}$, и открытый ключ $\{p, q, b\}$, а затем выполняет проверку электронной подписи. Она хеширует полученное сообщение таким же алгоритмом и получает $h = 19$, а затем проверяет сравнение

$$4^2 \cdot 2^1 \equiv 9^{19} \pmod{17} \Rightarrow 32 \equiv 9^{19} \pmod{17},$$

которое оказывается верным, а значит сообщению стоит доверять.