ЛЕКЦИЯ 8

СХЕМА ЭЛЬ – ГАМАЛЯ. ИНФРАСТРУКТУРА ОТКРЫТЫХ КЛЮЧЕЙ

Схема **Эль – Гамаля** — это в некотором роде устаревший стандарт на электронную цифровую подпись. Он сравнительно прост для рассмотрения.

1. Генерация открытого и закрытого ключей в схеме Эль – Гамаля

Схема Эль – Гамаля основывается на задаче возведения в степень. Обычно выбирается большое простое число p и рассматриваются все операции в поле (или в мультипликативной группе) по модулю числа p. Выбирается случайное число g, которое является генератором (**генератор мультипликативной группы** — это элемент, возводя который во все степени можно получить все элементы группы). В данном случае все числа, которые взаимно просты с p, будут являться генераторами.

Далее вычисляется число y:

$$y = g^x \mod p.$$

Тогда открытым ключом будет являться набор (y, g, p), а закрытым — (x, g, p).

Сложность восстановления закрытого ключа по открытому каналу связана с так называемой задачей дискретного логарифма. Допустим, злоумышленнику известен открытый ключ (y, g, p):

$$g^x \mod p.$$

Чтобы восстановить x из этого выражения, нужно взять дискретный логарифм:

$$x = \log_g y \mod p.$$

Данная задача является настолько же сложной, насколько и задача факторизации. Не существует эффективных полиномиальных алгоритмов для вычисления числа x, хо-

тя существуют субэкспоненциальные алгоритмы и алгоритмы для квантового компьютера, которые якобы способны решать эту задачу.

2. Шифрование в схеме Эль – Гамаля

В этой схеме шифрование осуществляется другим образом, нежели в RSA. В случае RSA использовалось возведение числа в степень. В рассматриваемой схеме имеется промежуточное дополнительно-случайное число k.

Выбирается случайное число k, находящееся в пределах от 1 до $(p-1)$, взаимно простое с $(p-1)$, после чего вычисляется число a:

$$a = g^k \mod p.$$

Далее вычисляется число b:

$$b = y^k M \mod p.$$

Подписью является набор из двух чисел: a и b. Соответственно, по каналу связи передаётся, во-первых, само сообщение M, во-вторых, его подпись — a и b.

Для любых систем с открытыми ключами по каналу связи передаётся не подпись от сообщения (её размер слишком большой). Предварительно вычисляется хеш-функция $h = H(M)$ от сообщения, и тогда подпись вычисляется уже от хеш-функции сообщения, потому что хеш-функция имеет фиксированные размеры, и за счёт этого электронно-цифровая подпись тоже будет малого размера.

В случае с шифрованием выбирается случайный ключ, который называется сессионным ключом S. Этот сессионный ключ зашифровывается схемой Эль – Гамаля, а закрытый текст получается путём шифрования сообщения M на выбранном случайном ключе.

Тогда передаваться будет зашифрованный сессионный ключ и зашифрованное сообщение. При этом, если сессионный ключ имеет размер 128 бит, то размер зашифрованного ключа будет всё равно не меньше 256 бит.

На самом деле, так как число p выбирается не меньше 1000 бит длиной, тогда размер зашифрованного сообщения будет 2000 бит. В схеме Эль – Гамаля всегда размер зашифрованного сообщения в 2 раза больше, чем размер исходного, за счёт того, что число k было выбрано случайным образом.

При каждом шифровании получается новый результат, то есть не повторяется та проблема, которая была у системы «ванильной RSA». В RSA одинаковые сообщения при шифровании давали одинаковый выход. Здесь так не получится: одинаковые сообщения, если у них будут разные числа k, будут давать разные числа a и b на выходе.

3. Расшифрование в схеме Эль – Гамаля

В этой схеме расшифрование проходит быстро:

$$M = \frac{b}{a^x} \mod p.$$

Доказательство корректности:

$$a^x = g^{kx} \mod p,$$

$$\frac{b}{a^x} = y^k \frac{M}{a^x} = \frac{g^{kx} M}{g^{kx}} = M \mod p.$$

Эта криптосистема очень проста с математической точки зрения.

Восстановление сообщения без знания ключа происходит следующим образом: злоумышленник знает открытый ключ (g, p, y) и числа a и b, которые были переданы по каналу связи. Для восстановления нужно найти

$$M = by^{-k} \mod p,$$

где

$$k = \log_g a \mod p.$$

Для нахождения k нужно вычислить дискретный логарифм. Оказывается, что задача вычисления дискретного логарифма не проще, чем задача по восстановлению ключа. В криптосистеме RSA это были две разные задачи: в одном случае это был дискретный корень степени e. Для восстановления ключа использовалась факторизация.

Возникает одна и та же проблема (проблема дискретного логарифма) и для восстановления закрытого ключа из открытого, и для попыток восстановления сообщения из переданного текста.

4. Цифровая подпись в схеме Эль – Гамаля

В этой схеме тоже вычисляются два числа, но сложнее. Сначала выбирается случайное число k. Затем считаются числа

$$a = g^k \mod p,$$

и

$$b = (M – xa) * k^{-1} \mod (p–1).$$

Проверка подписи выполняется следующим образом: когда к получателю приходит сообщение, ему известен открытый ключ и числа a и b. Нужно сравнить числа $y^a a^b \mod p$ и $g^M \mod p$. На самом деле, получатель вначале считает хеш-функцию от сообщения M и производит вычисления:

$$y^a a^b \mod p = g^h \mod p.$$

Система Эль – Гамаля считается устаревшей из-за того, что, все вычисления, которые выполняются в ней, производятся по модулю числа p или по модулю числа n. Эти мультипликативные группы хорошо изучены, и с ними можно эффективно работать.

Одной из самых больших потенциальных атак на любую криптосистему на эллиптических кривых является сведение её к изоморфной системе типа Эль – Гамаля. В криптосистеме Эль – Гамаля используется некое поле или некая мультипликативная группа G. В ней выбирается генератор x, причём генератор принадлежит группе G и $x < |G|$.

Затем выбирается некий элемент $y = g^x$ в группе G. Если рассматривать G по основанию p, то получается криптосистема Эль – Гамаля. Если же рассмотреть G как группу точек эллиптической кривой, то получится криптосистема на эллиптических кривых (только в этой группе происходит не умножение, а сложение).

Может возникнуть задача:

$$B = xA.$$

Рассмотрим число A, секретное число x и точку B, которая равна точке A, сложенной с собой x раз. B является открытым ключом, а x — закрытым. A — тоже часть открытого ключа. Это генератор, или хотя бы генератор достаточно большой подгруппы из G.

Если известны A и B, то можно найти x только с помощью полного перебора (потому что не существует операции деления одной точки на другую). Получается, что криптосистема на эллиптических кривых — это и есть криптосистема Эль – Гамаля по своей основе, но в другой группе.

Если в группе точек эллиптической кривой количество элементов является простым числом $(p–1)$, то за счёт изоморфности всех мультипликативных групп можно свести группу точек эллиптической кривой просто к группе чисел от 1 до $(p–1)$, после чего задача деления одной точки на другую сводится к задаче дискретного логарифма, которую хотя всё ещё нельзя эффективно решить полиномиальным алгоритмом, но субэкспоненциальные алгоритмы известны.

Одна из самых больших потенциальных атак на криптосистему на эллиптических кривых — это анализ структуры группы точек эллиптической кривой и сведение её к известной группе точек, операции в которой уже будут чисто математическим сложением и умножением, то есть, когда их можно будет записать в одну строчку, что гораздо проще, чем работать с формулами для сложения точек эллиптической кривой.

5. Инфраструктура открытых ключей

Системам с открытыми ключами свойственны две больших уязвимости. Во-первых, криптосистемы основаны на математических проблемах, для которых неизвестно, есть решение или нет. Например, в случае с криптосистемами на открытых ключах пока не найдено эффективных решений, но это не означает, что их нет. Также существуют различные сложные проблемы: проблема факторизации, проблема дискретного логарифма, проблема дискретного логарифма в произвольной мультипликативной группе.

Вторая проблема связана с необходимостью надёжного открытого канала, в котором злоумышленник не имел бы возможности что-то поменять **(идеальный канал)**.

В случае криптосистемы на закрытых ключах факт того, что получатель смог расшифровать сообщение, означает, что шифровкой занимался тот, кто обладает секретным ключом. В случае же криптосистем на открытых ключах тот факт, что у получателя получилось расшифровать сообщение, ничего не значит, потому что зашифровать сообщение может кто угодно.

5.1. Атака «человек посередине»

Другая проблема связана с тем, что неизвестно, кому принадлежат открытые ключи. Если собеседники A и B обменялись между собой открытыми ключами, это ещё не

значит, что они получили открытые ключи друг друга. Более того, если они получили открытые ключи, это ещё не означает, что эти ключи принадлежат, соответственно, A и B. Посередине могли встать злоумышленники и сгенерировать ключи.

Нужно сделать так, чтобы, используя открытый канал, всё-таки можно было установить надёжную связь и быть уверенным в том, что текст передаётся именно тому человеку, которому надо.

Основная идея состоит в использовании так называемого **промежуточного центра** (доверенного центра). В доверенный центр T (Трент) заранее обращается каждый из собеседников с просьбой подписать открытый ключ. И тогда пакет документов, в котором приведены справочные данные об одном их собеседников с подписью открытым ключом Трента, называется **сертификатом открытого ключа**. Такой сертификат находится у обоих собеседников для их ключей. Также у обоих собеседников есть сертификат Трента, как доверенного центра.

Если собеседники хотят общаться, они обмениваются своими сертификатами и проверяют, корректно ли подписан сертификат подписью Трента. Если корректно, то этот открытый ключ верен, и он действительно принадлежит тому, кто указан в сертификате. В таком случае возникает проблема однофамильцев, но она просто решается добавлением в сертификат даты рождения, и найти двух человек с одинаковой датой рождения становится сложно. Также можно в сертификат включить электронную почту.

В современном мире существуют сертификаты двух типов: государственные (они выдаются на паспорт) и сертификаты, которыми принято обмениваться в Интернете (они выдаются на электронную почту).

Существует два вида систем с доверенными центрами.

5.2. Система корневых доверенных центров сертификации

В этой системе используется набор центров сертификации: от T_1 до T_n. Когда пользователь приходит в один из центров сертификации, некий доверенный центр T_x раздаёт подписи промежуточным узлам. К некоторым из них можно обратиться с просьбой подписать ключ.

Также может быть основной центр сертификации (**корневой центр сертификации**), который, например, подписывает открытый ключ директору предприятия, дальше директор подписывает начальникам отделов, начальники отделов подписывают уже ключи работников.

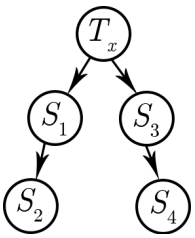

Рис. 8.1

Когда собеседники общаются, они отправляют друг другу цепочку сертификатов: сертификат корневого центра, промежуточные сертификаты и свой сертификат, подписанный по цепочке. Соответственно, получатель проверяет цепочку сертификатов на то,

что подписи валидны. Кроме того, сертификаты обладают сроком действия: 2 года, 3 года, 5 лет.

Более того, получатель проверяет цель выдачи сертификата. Например, сертификат может быть выдан на подписывание электронной почты, на подпись исполняемых файлов программ, на то подпись чужих сертификатов.

Часто Трент с целью получения выгоды организует выдачу сертификатов с возможностью подписывать другие сертификаты за значительно большую сумму денег, чем за сертификаты без такой возможности.

Получатель проверяет, что сертификат Трента находится в списке доверенных корневых сертификатов (у каждого человека есть свой список доверенных центров сертификации). Этот список обычно поддерживается программным обеспечением. Когда пользователь с помощью браузера заходит на защищённый сервер, где используются криптосистемы на открытых ключах, в каждом браузере есть набор доверенных центров сертификации. С каждым обновления браузера список этих центров сертификации тоже обновляется.

Недостаток подобного подхода состоит в том, что никто от подобного не застрахован. Доверенных центров сертификации много, и любой из них может выдать сертификат любому узлу в Интернете.

5.3. Схема с сетью доверия (PGP)

Это набор программ, соглашений и стандартов по обмену зашифрованными и подписанными сообщениями. Эта программа была впервые написана **Филом Циммерманом** примерно в 1995-м году. Он предполагал её продавать, но оказалось так, что в тот момент правительство США решило внедрить на законодательном уровне так называемую инициативу Clipper. Эта инициатива обязывала всех производителей криптографического аппаратного и программного обеспечения внедрять в свои системы так называемый backdoor — специальную микросхему от правительства США.

Данная микросхема обладала таким шифром, который очень легко вскрыть при наличии специального шифра, депонированного в бюро шифров, который хранится в специальном хранилище. Предполагается, что доступ к этому хранилищу выдается только по судебному распоряжению.

Чтобы противостоять такой инициативе, Фил Циммерман выпустил свою программу как open source, то есть в открытом виде. Более того, он решил её экспортировать за пределы США. Но алгоритм шифрования нельзя использовать, если у создателя подписан контракт с государственной организацией.

Например, производство на продажу любых криптосистем в России является лицензируемой деятельностью, причём процедура сертфикации секретна. Считается, что это требуется для того, чтобы все криптосистемы были достаточно надёжными.

Но PGP было открытой программой. Исходный код программы был опубликован в Интернете, якобы только в США. Эту программу распечатали, вывезли как книгу, отсканировали, скомпилировали и добились работающей программы.

В то время, когда PGP экспортировался, максимально разрешённые длины ключа для экспортируемых криптосистем составляли 48 бит. В PGP длины ключей были: 128 бит для криптосистем на закрытых ключах и 1000, 2000, 4000 бит — для криптосистем на открытых ключах.

Изначально в PGP хотели реализовать другую идею: открытый ключ может подписать любой человек. Каждый участник подписывает своим ключом сертификаты других участников, которых он знает лично или которых он видит перед собой. В **open source комьюнити** регулярно устраиваются так называемые public key party, то есть «вечеринки публичных ключей»: каждый участник приходит туда лично, знакомится с другими участниками и просит подписать его открытый ключ. Такого рода «вечеринки» обычно устраиваются на международных конференциях; ключи подписываются на международном уровне. За счёт этого получается сеть доверия.

После этого ключи загружаются на специальный сервер. Получатель запрашивает ключ отправителя, указывая его e-mail адрес и начинает наблюдать, кто подписал этот открытый ключ. Дальше он получает список людей, которые подписывали эти ключи — так сеть доверия увеличивается. В конце концов он получает одно или несколько пересечений со своей сетью доверия, смотрит, верна ли цепочка подписей и насколько длинна эта цепочка (потому что если цепочка состоит из 30-ти сертификатов, то, скорее всего, никакого доверия быть не может, а если цепочка достаточно короткая (3-4 сертификата), то, наверное, имеет смысл доверять).

5.4. Уровни доверия

В этой схеме можно выбрать три **уровня доверия**:

1. Разрешить подписать ключ, если знать человека лично;
2. Разрешить подписать ключ и разрешить ключу подписывать другие ключи, будучи уверенным, что человек умеет правильно выбирать достойных подписи людей;
3. Разрешить подписать ключ и разрешить ключу подписывать другие ключи, будучи уверенным, что человек будет подписывать ключи только тем людям, которые, в свою очередь, будут проверять чужие ключи по паспорту.

Эти уровни доверия записываются в сертификатах.

Также существует абсолютный уровень доверия (четвёртый), который свойственен только одному ключу — личному. У всех остальных ключей есть градация уровня доверия, и цепочка обычно может состоять не более чем из четырёх элементов.