Отчет по прохождению внешнего курса.

Раздел 3. Криптография на практике.

Рогожина Надежда Александровна

Содержание

Список иллюстраций

Список таблиц

1 Цель работы

Цель данного курса - узнать, как обеспечивается безопасность интернет-трафика, какие пароли нужно выбирать и как их хранить, познакомиться с методами защиты сообщений в мессенджерах (WhatsApp, Telegram), понять, как работают механизмы аунтефикации в электронных платежах, а также зачем нас иногда просят выбрать квадраты, где изображены светофоры.

2 Выполнение

2.1 Введение в криптографию

1. В асимметричных криптографических примитивах (рис. [2.1]):

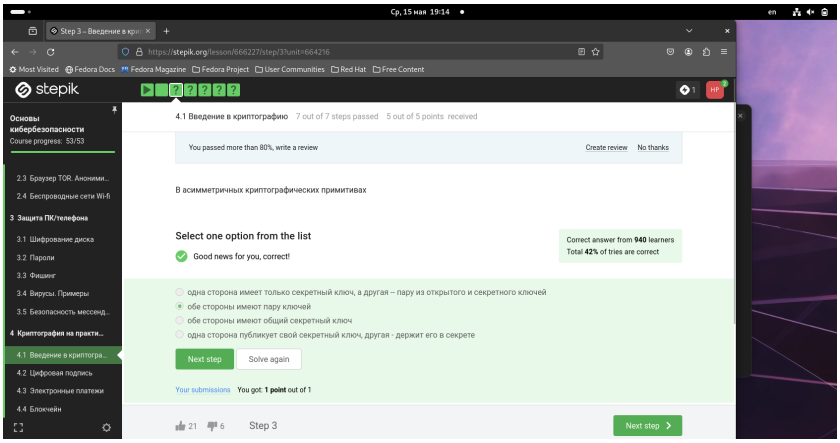

Рис. 2.1: Ответ

Определяющее свойство симметричной криптографии состоит в том, что она включает себя протоколы, где две или более стороны имеют общие секретные ключи, поэтому она и называется симметричной. К таким протоколам относят симметричное шифрование и некоторые протоколы аутентификации. Часто симметричный протокол довольно сложно построить, сложно установить потенциальный канал связи, исключительно основываясь на симметричных протоколах, поскольку нам нужно сгенерировать общий секретный ключ, то есть либо как-то физически встретиться с другим человеком и с другим устройством, либо что-то

такое сделать, чтобы мы сгенерировали общий секрет. И элегантным решением этого вопроса являются протоколы асимметричной криптографии.

2. Криптографическая хэш-функция (рис. [2.2]):

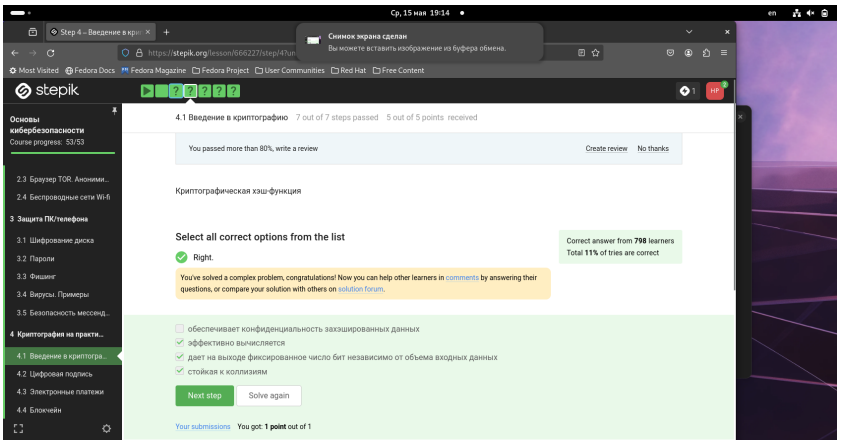

Рис. 2.2: хэш-функция

Криптографическая хэш-функция берет на вход произвольный объем данных, то есть какие-то биты и выдает на выходе фиксированную строку, например длины n. Важно, что, как правило, функция сжимает данные: она берет большой набор данных и выдаёт потом маленькое фиксированное значение. Важное свойство криптографической хэш-функций, то, что делает её криптографической – это стойкость к коллизиям. Что такое коллизия? Коллизия – это два разных входа в хэш-функцию, которые дают одинаковый выход. То есть это две разные строки например x и x’, где $x \neq x'$, такие, что значения хэш-функции на них совпадают, то есть $h(x) = h(x')$. Это важное свойство отличает криптографическую функцию от некриптографической. Можно доказать (мы этого делать с вами не будем), что из этого свойства коллизии следует другое важное свойство, а именно то, что криптографическую хэш-функцию сложно обратить. То есть, если я вам даю какое-то значение этой функции в точке h(x) и спрашиваю вас, как найти x, то есть вход в эту функцию, для современных криптографических хэш-функций это сделать сложно. Благодаря этим свойствам, криптографические функции

широко применяются в коммуникациях, мы с вами в одной из лекций говорили о том, как криптографическую хэш-функцию можно использовать для хранения паролей. Она также используются для протоколов, подтверждающих целостность данных, ну и современное довольно популярное применение хэш-функции – это доказательство работы. По-другому это называется протоколом proof of work, который используется, например, в таком блокчейне, как биткойн.

3. К алгоритмам цифровой подписи относятся (рис. [2.3]):

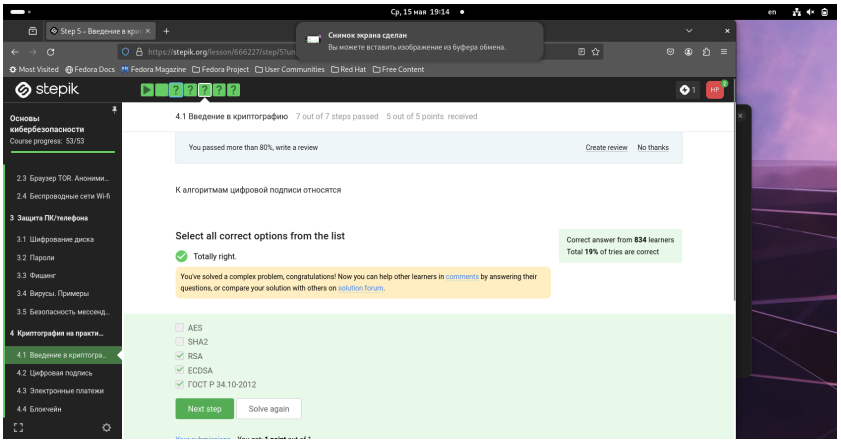

Рис. 2.3: ЦП

К примерам цифровой подписи относятся интернет-сертификаты, подпись RSA, американский стандарт ECDSA и отечественный стандарт ГОСТ стандарт Р 34.20.2012.

4. Код аутентификации сообщения относится к (рис. [2.4]):

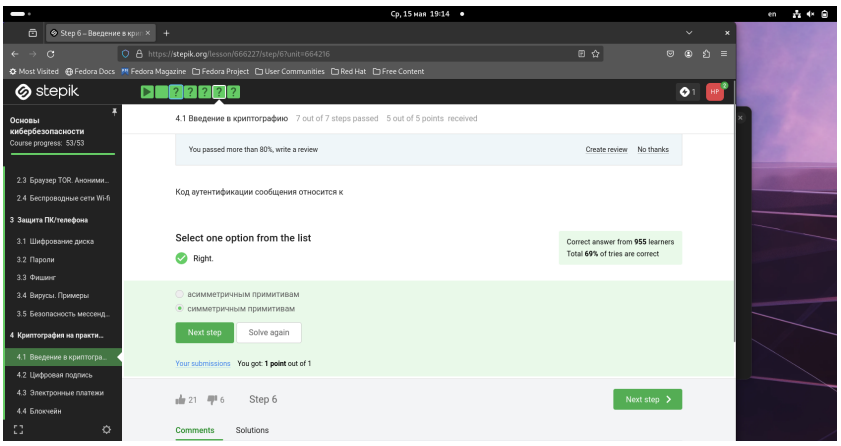

Рис. 2.4: ответ

Если мы будем двигаться дальше, то симметричное шифрование можно проапгрейдить до симметричного шифрования с аутентификацией. Это означает, что к шифр-тексту, который мы сгенерировали с помощью ключа для какого-то сообщения, мы еще добавляем код аутентификации сообщения. Это также симметричный примитив, который берет на вход какой-то ключ (это должен быть другой ключ, не тот, с которого мы шифровали) и сообщение и выдает код аутентификации сообщения. Корректно об этом примитиве думать, как о симметричной версии подписи. Как правило, код аутентификации сообщения строится с помощью хэш-функции или симметричного шифрования.

5. Обмен ключами Диффи-Хэллмана - это (рис. [2.5]):

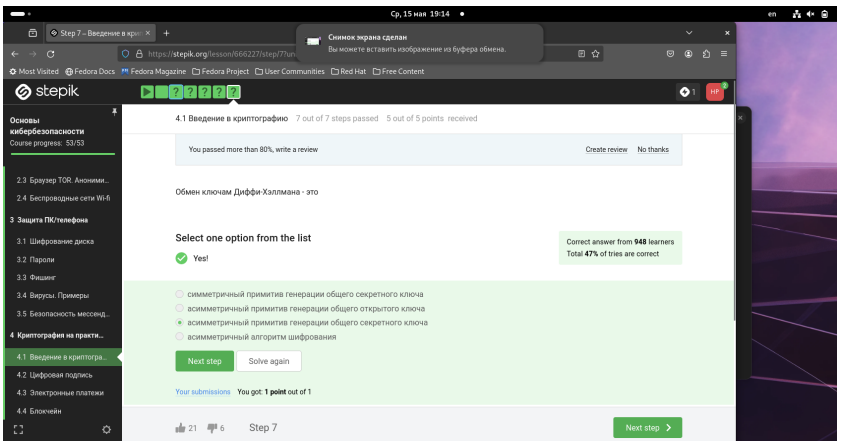

Рис. 2.5: Диффи-Хэллман

Самым популярным примером протокола обмена ключами является протокол Диффи-Хэллмана, как раз он, либо его модификации используются в современных мессенджерах и в протоколе TLS для того, чтобы мы смогли сгенерировать общий секретный ключ и дальше шифровать наши данные с помощью симметричного алгоритма, то есть с помощью ключа skAB. Если реализовать генерацию общего ключа так, как она описана у Диффи-Хэллмана, мы получим довольно слабый протокол, нестойкий к активным злоумышленникам. Сделать этот протокол стойким к активным злоумышленникам помогает цифровая подпись.

2.2 Цифровая подпись

1. Протокол электронной цифровой подписи относится к (рис. [2.6]):

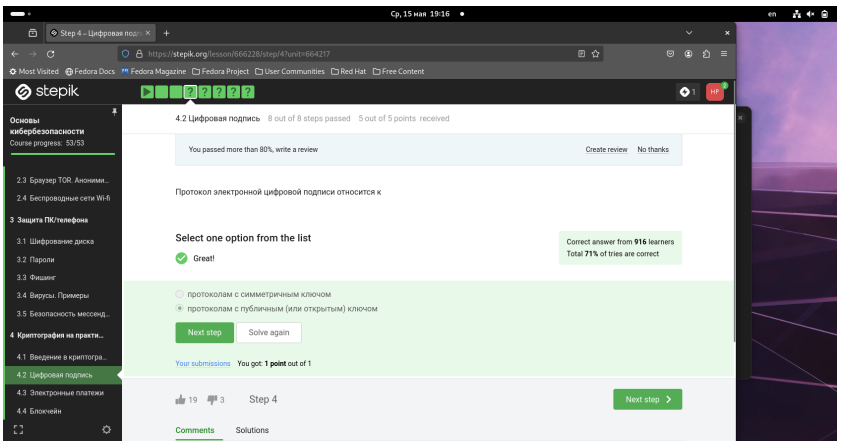

Рис. 2.6: ответ

Люди, которые занимаются разработкой программного обеспечения, с выходом каждого нового обновления или с выходом самой первой версии программы обязательно подписывают программный код. Например, если вы хотите обновить свою операционную систему, допустим, Ubuntu, у организации программистов, которые выпустили обновление, есть пара ключей – это публичный ключ pkU и секретный ключ skU. Секретный ключ они держат в секрете, публичный ключ знают все. И когда они выпускают обновление, они его подписывают с помощью своего секретного ключа.

2. Алгоритм верификации электронной цифровой подписи требует на вход (рис. [2.7]):

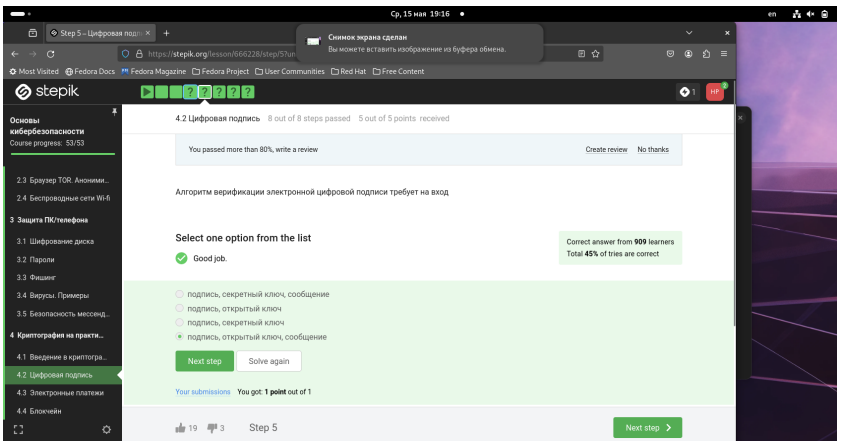

Рис. 2.7: Алгоритм

Алгоритм подписи Sign берет на вход секретный ключ и обновление, например, программный код и выдает подпись ⊠. Обновление вместе с подписью скачивается всеми пользователями, и, прежде чем установить это обновление, ваш компьютер проверяет, действительно ли это обновление пришло от тех программистов, от которых мы ожидаем это обновление, что это не поддельный софт. То есть каждая машина запускает процедуру Verify, которая берет на вход само обновление, подпись и открытый ключ разработчика, и в случае если верификация прошла успешно, мы можем установить это обновление.

3. Электронная цифровая подпись не обеспечивает (рис. [2.8]):

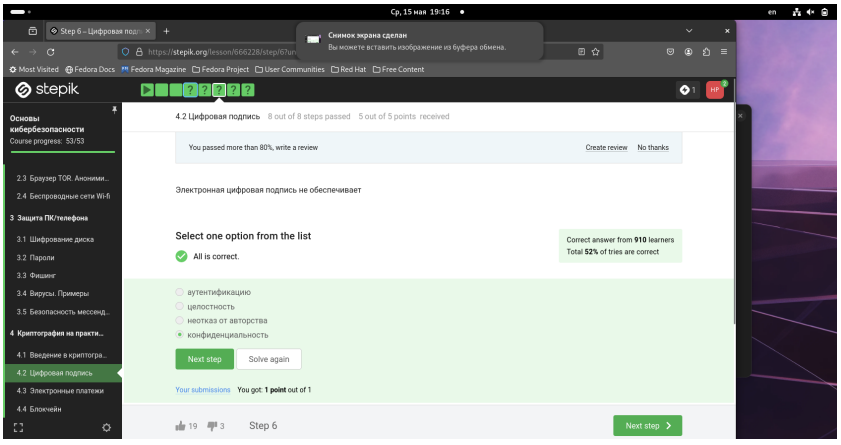

Рис. 2.8: ЦП

В алгоритме подготовки и верификации цифровой подписи участвуют как минимум 3 узла, поэтому, о конфиденциальности говорить сложно.

4. Какой тип сертификата электронной подписи понадобится для отправки налоговой отчетности в ФНС (рис. [2.9]):

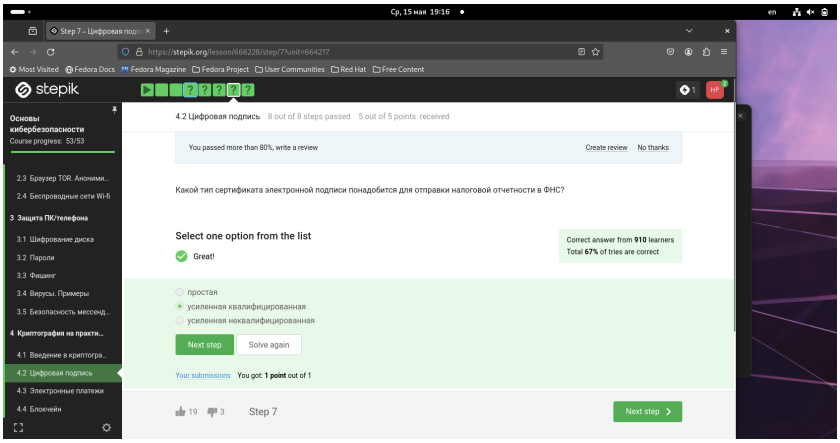

Рис. 2.9: Сертификат

Существует три различных точки зрения на подписи: простая, усиленная неквалифицированая и усиленная квалифицированная. Первые два типа не имеет юридической силы или она довольно ограничена, но их можно сгенерировать у себя на компьютере. Простую мы генерируем с помощью пароля, PIN-кода, такая простая подпись может быть использована для авторизации на сайтах. Усиленная неквалифицированная подпись может быть подтверждена сертификатом, который вы сами же можете и выпустить, то есть кроме того, что вы выпускаете свою пару секретных ключей, вы можете сами их сертифицировать. Такая подпись может быть использована в коммерческом документообороте в небольших негосударственных структурах. А вот что касается усиленной квалифицированной подписи, эта подпись уже имеет юридическую силу, она, как правило, равнозначна рукописной. Для того, чтобы получить такую подпись, вам нужно пойти со своим паспортом и с другими данными в сертификационный центр, который

должен быть аккредитован конкретным министерством. Такие подписи используются на Госуслугах, в государственном документообороте. Подробнее об этом написано в федеральном законе № 63.

2.3 Электронные платежи

1. Выберите из списка все платежные системы (рис. [2.10]):

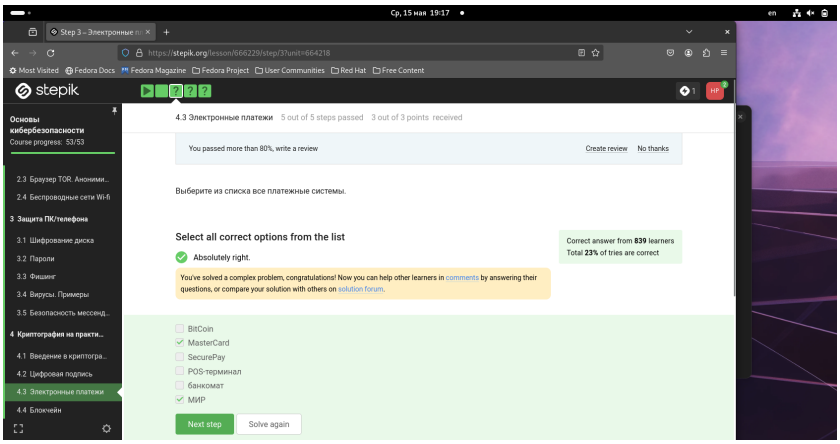

Рис. 2.10: ПС

Из всех перечисленных вариантов, только MasterCard и МИР являются платежными системами. Также есть, например, VISA.

2. Примером многофакторной аутентификации является (рис. [2.11]):

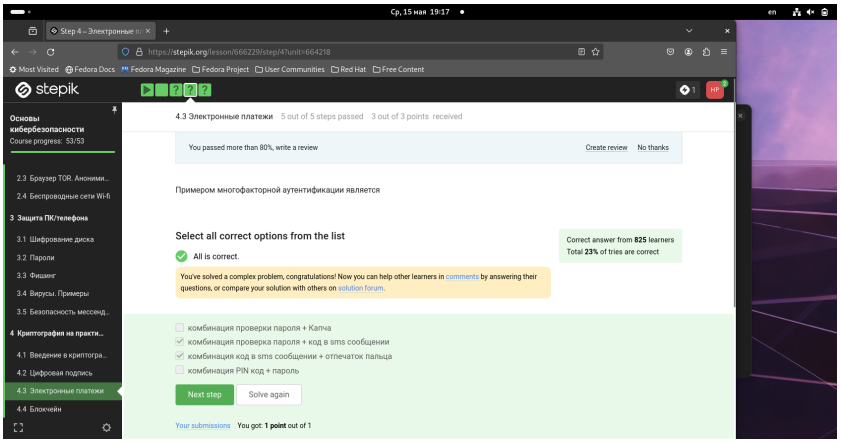

Рис. 2.11: МА

Многократная аутентификация заключается в том, что мы доказываем в ходе этого протокола несколько вещей есть. Основные категории вещей, которые мы можем доказать: 1) то, что я знаю – это либо пароль, либо PIN-код, либо в случае онлайн-платежей это секретный код, 2) конкретно в онлайн-платежах мы еще используем второй фактор – это то, чем я владею, например, телефон, именно поэтому нам часто приходит код, который вы должны подтвердить или вбить в ваш браузер, 3) другой фактор аутентификации – это свойства, например, биометрия, опечаток пальца, сетчатки глаза, 4) четвертый фактор аутентификации – локация. Способ аутентификации, как правило, выбирается банком.

3. При онлайн платежах сегодня используется (рис. [2.12]):

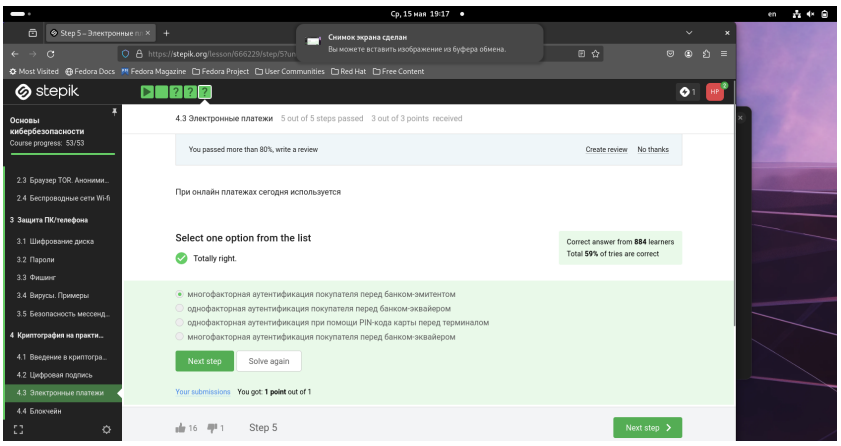

Рис. 2.12: Онлайн платежи

Сейчас однофакторная аутентификация покупателя практически не используется, так как не имеет той степени безопасности, и в многофакторной аутентификации - подтверждать, что именно вы совершаете платеж необходимо банку-эмитенту.

2.4 Блокчейн

1. Какое свойство криптографической хэш-функции используется в доказательстве работы (рис. [2.13]):

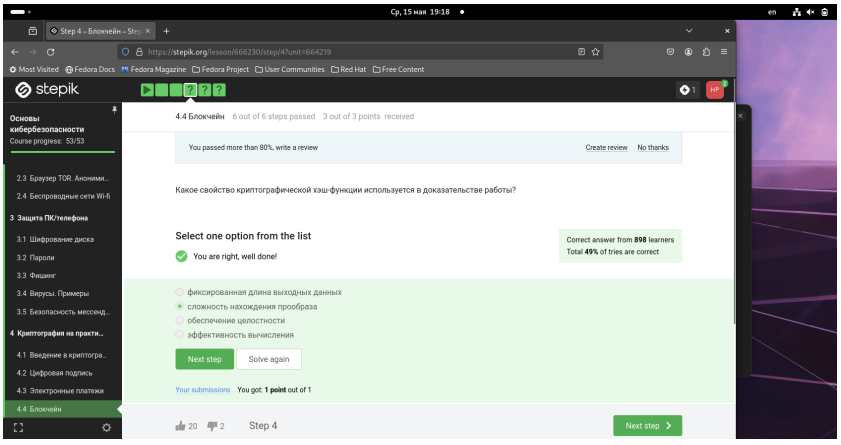

Рис. 2.13: свойство хэш-функции

Хэш-функция - это функция, которая берет на вход любые данные и выдает за какое-то быстрое время фиксированное число бит. И задача майнера в доказательстве работы - это отыскать такой вход в хэш-функцию, что ее значение имеет определенный паттерн, иными словами, отыскать такой x, что h(x) имеет, например, 17 первых нулей или 17 первых единиц, это неважно. В биткоине используют 18 или 19 первых нулей. Это число на самом деле может быть модифицировано относительно производительности сети в тот или иной момент времени. Оно примерно находится в районе в 17-19. Почему эта задача сложная? Потому что мы знаем, что для криптографической хэш-функции неизвестно на сегодняшний день атаки быстрее, чем перебором для конкретно этой задачи. Если я вам даю какую-то хэш-функцию, например, SHA-3 или ГОСТ и прошу мне найти такой x, что h(x) содержит 17 первых нулей, и все, что вы будете делать - это перебирать разные x, смотреть на их выход и проверять, действительно ли выходное значение содержит 17 первых нулей или нет. Таким образом, ожидаемое число различных входных данных x, которое нам необходимо перебрать, пока мы не найдем нужный паттерн, примерно равно 217.

2. Консенсус в некоторых системах блокчейн обладает свойствами (рис. [2.14]):

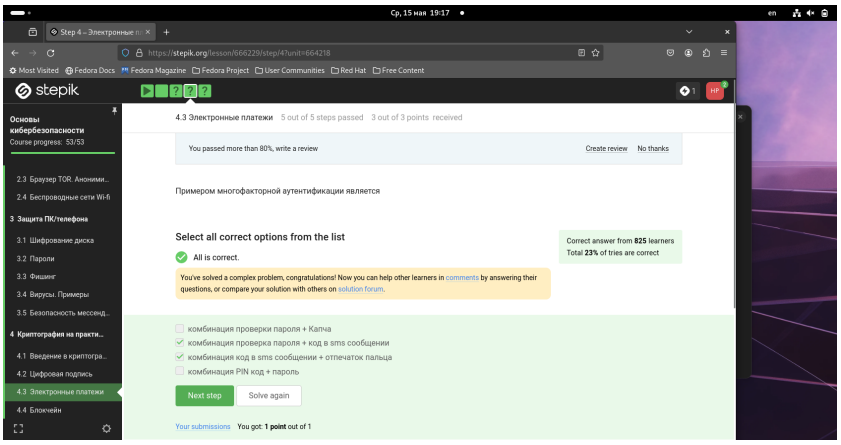

Рис. 2.14: консенсус

В основе любого блокчейна, в частности биткоина, лежит консенсус – соглаше-

ние, в терминах криптовалют консенсус - это некая публичная структура данных или ledger (переводится с английского как «бухгалтерская книга»), где просто содержится история всех переводов, хранится список того, кто что кому заплатил, в какое время. Почему консенсус? Потому что эта публичная структура, и бухгалтерский учет должен обеспечивать четыре основных свойства. Первое - это постоянство, то есть когда-либо добавленНые данные не должны быть удалены из этой структуры. Второе - это сам консенсус, то есть все участники видят одни и те же данные и соглашаются с одним и теми же данными, исключением могут быть последние пары блоков, то есть последние изменения в этом блокчейне, в этой публичной структуре данных. Третье - это живучесть, это означает, что мы можем добавлять новые транзакции, когда хотим, мы можем осуществлять платежи, когда хотим. И последнее четвертое свойство - это открытость, то есть любой человек может быть участником блокчейна. Это справедливо не для всех блокчейнов, для биткоина это справедливо.

3. Секретные ключи какого криптографического примитива хранят участники блокчейна (рис. [2.15]):

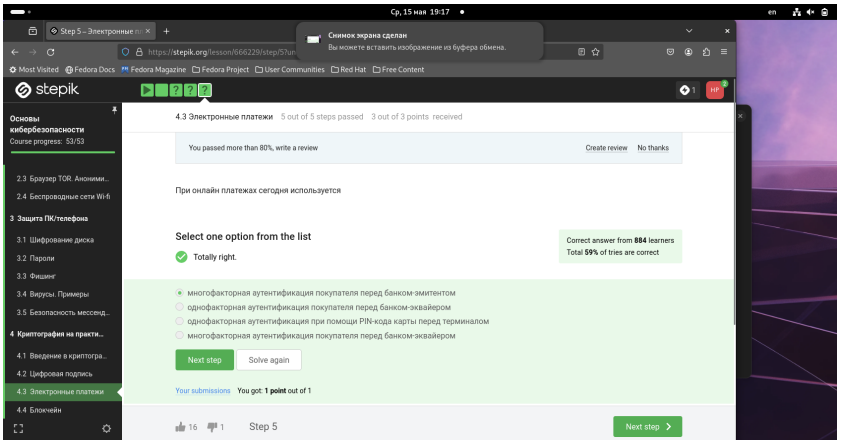

Рис. 2.15: примитив

Допустим, у нас вами есть в блокчейне 3 участника, которые обмениваются друг с другом транзакциями. Важно то, что у каждого участника есть свой секрет-

ный ключ, и своим секретным ключом мы всегда будем подтверждать какую-то транзакцию. Важно то, что этот ключ у нас секретный, мы его используем для подписи. Подпись – это и есть подтверждение моей транзакции. Мы с вами разбирали в одной из лекций, как работает электронная цифровая подпись, у этого примитива есть секретный и открытый ключи, и наш секретный ключ - это то, что позволяет нам совершать транзакции от нашего лица. То есть, если мы говорим, что мы в транзакции переводим 10 единиц денег какому-то человеку и подписываем эту транзакцию, это не может сделать никто, кроме нас, поскольку у нас есть секретный ключ. Допустим, у нас есть три участника сети, которые осуществили 3 какие-то транзакции и подписали их: первый со своим секретным ключом, второй - со своим и третий – со своим. Эти три транзакции формируются в один блок. Этот блок должен быть каким-то образом добавлен в сеть блокчейн. Кто из майнеров будет добавлять этот блок, решает как раз таки протокол консенсуса. Допустим, протокол выбрал верхнего майнера (мы далее поговорим, что это за протокол и что майнер должен сделать для того, чтобы быть выбранным), этот майнер формирует из трех транзакций блок, проверяет, что все там корректно, и добавляет этот блок в сеть. За это он получает награду: как только он совершил какую-то работу, то есть добавил блок в блокчейн, он получает некую награду за майнинг, состоящую из тех денег, которые добавляют вот эти три участника, которые инициировали транзакции.

3 Выводы

В ходе работы мы изучили, что такое криптография, цифровая подпись, как происходят электронные платежи и что такое блокчейн.

Список литературы