## Лекция 17. Криптографические методы защиты информации

Криптографические методы защиты позволяют реализовать следующие три составляющие системы информационной безопасности:

- конфиденциальность - не дать злоумышленнику прочесть передаваемую информацию
- целостность - не позволить ему изменять информацию так, чтобы об этом не узнал получатель
- аутентификацию - сделать так, чтобы злоумышленник не смог незаметно отправить информацию от имени другого лица.

Основой этих методов является *шифр (*или *криптосистема)* - обратимое преобразование открытых данных в зашифрованные.

Примером является популярный моноалфавитный шифр текстовых сообщений, основанный на замене букв на некоторые символы или другие буквы того же алфавита. В рассказе Артура Конана Дойля «Пляшущие человечки» Шерлок Холмс на основе частотного анализа букв и подбора текста взламывает такой шифр убийцы Аб Слени. Другим примером является *шифр Цезаря*, в котором каждая буква заменяется отстоящей от нее на 3 позиции буквой в алфавите: А - на Г, Б - на Д, и т.д. Вместо тройки можно использовать любой сдвиг направо или налево.

С распространением механических устройств шифрования, применявшихся для дипломатических и военных целей, возникла проблема попадания данных устройств в руки врага. В связи с этим в 1883 году голландский криптограф Огюст Керкгоффс сформулировал принцип криптографии, согласно которому секретность должна быть заключена не в алгоритме/механизме шифрования, а в используемом ключе. Тогда даже если враг завладеет криптосистемой, он не сможет расшифровать послания без ключа. Ключом называют некие параметры шифра. Например, для шифра Цезаря ключом является величина сдвига букв. Когда секретность системы (алгоритма) шифрования заключена только в ключе, ее можно опубликовать, сделать стандартом, если это программа, то открыть ее исходный код, чтобы другие криптографы смогли его проверить.

До середины 70-х годов прошлого века все криптосистемы для своей успешной работы требовали секретность ключа. О ключе общающиеся стороны были вынуждены договариваться заранее и использовать для его передачи альтернативные каналы связи. Однако в компьютерных сетях такой процесс сильно усложняет и замедляет работу. Почему бы не выполнять быструю

передачу ключа через сеть? Но как обеспечить секретность при передаче ключа шифрования? Чтобы сохранить конфиденциальность даже в случае, когда злоумышленник прослушивает весь трафик, были разработаны криптосистемы с открытым ключом. *Все старые шифры были симметричными* - в них ключ расшифровки совпадал или легко получался из ключа зашифровки. В новых асимметричных криптосистемах ключи зашифровки и расшифровки различны и, зная один, невозможно определить другой.

Асимметричные криптосистемы работают медленнее симметричных. Поэтому на практике данные шифруются симметричным алгоритмом, а шифрование передаваемого ключа симметричного алгоритма осуществляется асимметричной криптосистемой.

Рассмотрим ситуацию компьютер А хочет отправить компьютеру Б секретное сообщение (см. рис. 17.1). При использовании асимметричной криптосистемы перед передачей сообщения компьютер Б генерирует два ключа. Ключ зашифровки он отправляет компьютеру А и не боится, что его перехватит злоумышленник. Пусть перехватывает - расшифровать данные он ведь им не сможет. Этот ключ в данной ситуации называется открытым и может быть сделан известным всем (криптографы говорят “опубликован в газете”). Напротив, ключ расшифровки (закрытый ключ) компьютер Б хранит у себя в тайне и не передает по сети. После получения от Б открытого ключа компьютер А зашифровывает им сообщение и передает по сети. Для передачи данных в обратном направлении: от Б к А, открытый и закрытый ключи генерирует и компьютер А.

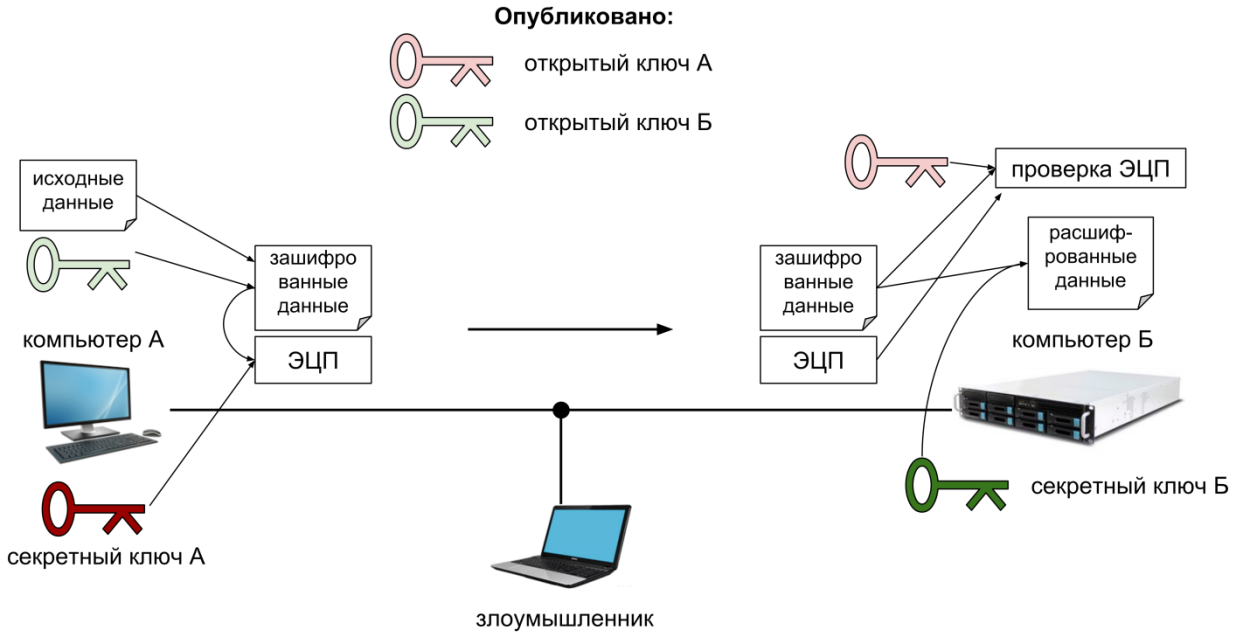


Рис. 17.1. Принципы работы асимметричной криптосистемы

Данная схема отлично работает в так называемом случае “пассивного злоумышленника”, который только прослушивает трафик, но не может изменять передаваемые пакеты или отправлять пакеты от своего имени. Активный же злоумышленник может перехватить пакет, идущий от А к Б, зашифровать открытым ключом свои данные и отправить их от имени А. *Как сделать так, чтобы компьютер Б понял, что злоумышленник подменил пакет?*

Для решения этой задачи на основе асимметричных шифров криптографы придумали *электронную цифровую подпись* (ЭЦП, Fingerprint - отпечатки пальцев). Чтобы создать ЭЦП для некоторых данных, нужно вычислить их хеш и зашифровать его асимметричным шифром с помощью некоторого секретного ключа, который называется секретным ключом создания ЭЦП.

Второй ключ - расшифровки ЭЦП - “публикуется в газете”, чтобы все смогли расшифровать, проверить подпись и убедиться, что это действительно данные настоящего отправителя. Таким образом, в схеме на рис.17.1 компьютер отправителя А после шифрования данных открытым ключом Б должен вычислить хеш зашифрованного пакета, зашифровать его своим закрытым ключом и отправить Б. Тогда отправляемый пакет будет “подписан”, и никто не сможет подделать эту подпись т.к. закрытый ключ А держит в секрете, но зато все смогут ее проверить, т.к. его открытый ключ “опубликован в газете”.

Чтобы активный злоумышленник не смог опубликовать поддельный открытый ключ проверки ЭЦП, его защищают с помощью цифрового сертификата, выданного удостоверяющим центром сертификации. Центру сертификации должны доверять отправитель и получатель, и оба должны иметь у себя на компьютерах данные этого центра сертификации, позволяющие удостовериться, что владелец ключа проверки ЭЦП - отправитель, а не злоумышленник. Сертификат включает в себя

- информацию о владельце
- его открытый ключ
- информацию о центре сертификации, выдавшем сертификат
- ЭЦП всего сертификата, зашифрованная с помощью закрытого ключа центра сертификации
- другие поля, например, время действия, тип (сертификат домена, центра сертификации и др.), алгоритмы шифрования и т.п.

Крупные организации часто нуждаются в выпуске или обновлении сертификатов для своих серверов, поэтому им удобнее организовывать свои

собственные центры сертификации. Хранить и регулярно обновлять на каждом компьютере данные всех существующих центров сертификации в мире невозможно, поэтому было решено упорядочить центры сертификации в виде иерархической схемы, на вершине которой находятся корневые центры сертификации. Этих центров немного и их данные должны храниться на всех компьютерах. Они “доверяют” центрам сертификации крупных организаций, т.е. подписывают своим закрытым ключом их сертификаты. Их данные уже не обязаны храниться на компьютерах пользователей. Эти центры сертификации (крупных организаций) “доверяют” отдельным серверам или пользователям т.е. выпускают для них сертификаты и подписывают их своим закрытым ключом . Возможно и большее количество элементов в этой цепочке доверяющих друг другу субъектов.

Когда клиент устанавливает соединение с сервером по криптографически защищенному протоколу, сервер отправляет в ответ всю цепочку сертификатов: его собственный сертификат, сертификат выдавшего этот собственный сертификат центра сертификации и корневого центра (см. пример на рис. 17.2).

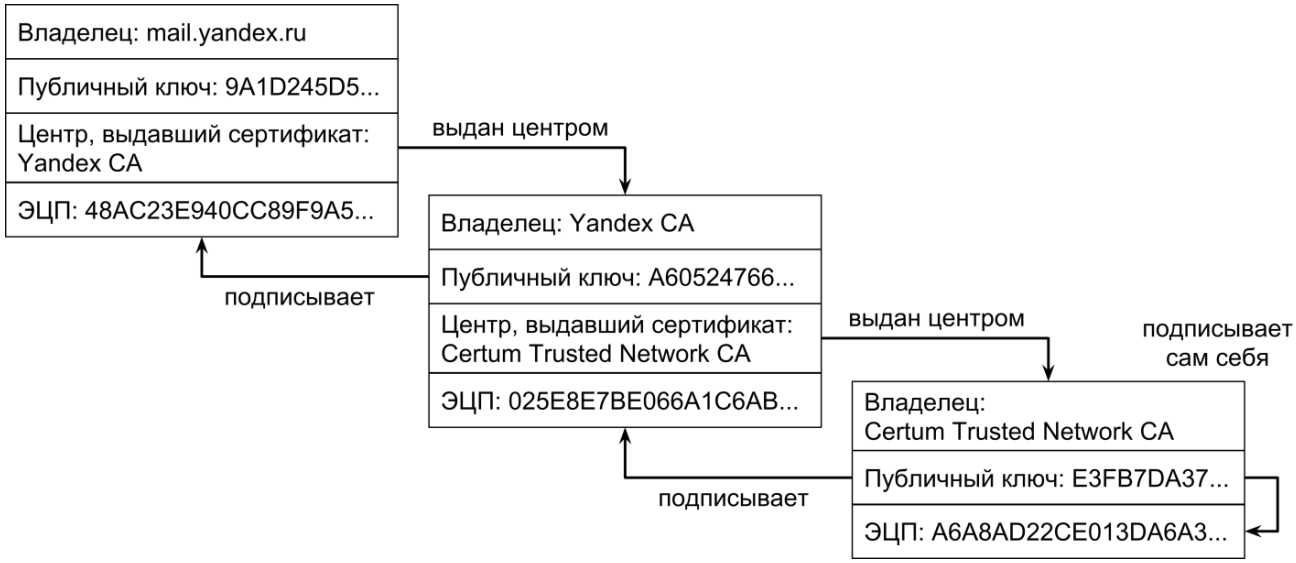


Рис. 17.2. Цепочка сертификатов сервера mail.yandex.ru

Клиент начинает проверять все сертификаты по очереди, начиная с сертификата сервера до корневого. Проверяются различные данные, включая ЭЦП с помощью открытого ключа следующего в цепочке сертификата (см. рис. 17.2). Последний сертификат должен совпасть с тем, который храниться на компьютере клиента. В случае ошибки на каком-нибудь шаге происходит отказ в установлении защищенного соединения.