1. Прикладные протоколы

*Построение семейства протоколов KriptoKnight на основе базовых протоколов взаимной аутентификации и распределения ключей. Особенности построения семейства протоколов IPsec. Протоколы Oakley, ISAKMP, IKE. Протоколы SKIP, SSL/TLS и особенности их реализации*

В простейшем случае криптографический протокол – это последовательность взаимодействий двух удаленных пользователей (которые не видят друг друга и не слишком друг другу доверяют), предназначенная для решения какой-либо простейшей задачи. Тем не менее, точное выполнение протокола заведомо гарантирует строгое выполнение поставленной задачи, даже если один из пользователей попытается обмануть другого. Такие криптографические протоколы называются примитивными.

Объектом прикладных криптографических протоколов являются удаленные абоненты, взаимодействующие, как правило, по открытым каналам связи. Целью взаимодействия абонентов является решение какой-то задачи. Имеется также противник, который преследует собственные цели. При этом противник в разных задачах может иметь разные возможности: например, может взаимодействовать с абонентами от имени других абонентов или вмешиваться в обмены информацией между абонентами ит.д. Противником может даже оказаться один из абонентов или несколько абонентов, вступивших в сговор.

Приведем еще пример задач, решаемых удаленными абонентами. Взаимодействуют два не доверяющих друг другу абонента. Они хотят подписать контракт. Это надо сделать так, чтобы не допустить следующую ситуацию: один из абонентов получил подпись другого, а сам не подписался. Протокол решения этой задачи принято называть протоколом подписания контракта.

Объединяя в нужной последовательности несколько примитивных протоколов, можно получить прикладные криптографические схемы, в которых участвует практически неограниченное число пользователей и которые предназначены для решения вполне серьезных и вполне актуальных задач, например, организации электронных торгов, или электронного голосования, или организации электронной платежной системы. Эти схемы можно классифицировать следующим образом.

* Протокол с арбитражем

Арбитр — это один из участников протокола, которому все остальные доверяют. Арбитр не преследует целей сторон, он независим. Существует ряд проблем, которые влияют на такой принцип системы. К примеру в реальной жизни роль арбитра берет на себя адвокат. Приведем аналогии.

Довериться адвокату легко, если про него много известно и у него отличная репутация. И при этом если два участника протокола не доверяют друг другу, врядли они захотят доверять анонимному участнику которого они не знают.

Адвокат за свои услуги берет конкретные материальные средства. Как и кому оплачивать за аналогичные услуги в компьютерной сети?

При ввода арбитра в протокол, увеличивается время работы протокола.

При возрастании арбитров, что бы уменьшить влияния арбитра на роботу протокола, увеличивается также время реализации и расходы на протокол.

Так как арбитр есть главным в работе протокола, то все атаки будут реализованы на него, и он есть слабым звеном в таком протоколе.

несмотря на недочеты, такой протокол имеет широкое применение в компьютерных сетях.

* Протокол с судейством

Для снижения расходов на арбитраж, функции арбитража делят на две части. Первая совпадает с функциями арбитража полностью. Вторая функции у еще одного участника протокола, судья. Функции похожи на функции арбитра, только он вступает в действие когда участники в определенном шаге начали сомневаться в правильности протокола. Также судья может выявить, кто ведет себя не честно.

* Самоутверждающийся протокол

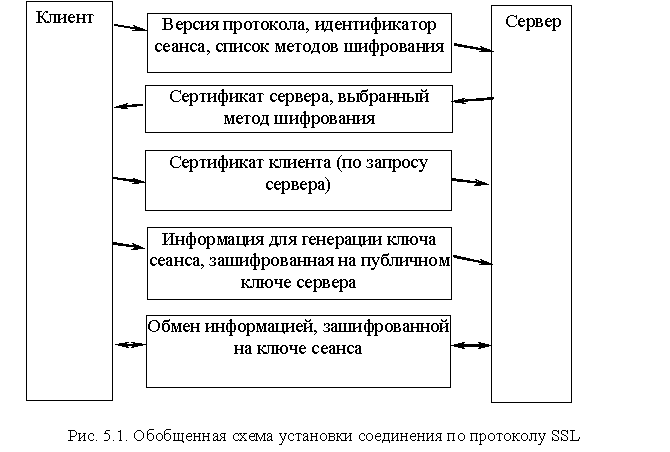
Это протокол который не требует арбитра для завершения каждого этапа протокола. Не предусмотрено также наличие судьи для решения конфликтных ситуаций. Этот протокол работает по принципу, что если кто-то начал мошенничать вовремя работы протокола, все остальные участники это увидят и могут прекратить дальнейшее использование протокола. Также нужно рассмотреть виды атак на криптографические протоколы. К примеру пассивной атаки — это подслушивание информации при транслировании. Активная атака предусматривает, что злоумышленник изменит ход роботы протокола в свою выгоду. Также есть протоколы с использованием симметричного шифрования и использованием шифрования с открытым ключом.

Для обеспечения конфиденциальности информации, передаваемой по сети, необходимо обеспечить ее шифрование на стороне отправителя и дешифрацию на стороне получателя. Самым простым средством является шифрование информации на прикладном уровне. В этом случае шифрованию подвергается только непосредственно передаваемая информация, никакая служебная информация из заголовков сетевых пакетов в этом случае не кодируется. Примером программы, которая осуществляет подобного рода шифрование можно назвать PGP (Pretty Good Privacy). Одно из главных достоинств этой программы состоит в том, что существуют версии PGP практически для всех программных платформ: DOS, Windows, Unix, Macintosh. PGP представляет собой криптосистему, которая позволяет шифровать данные (содержимое файлов, буфера обмена) по асимметричной схеме, а также формировать ЭЦП для передаваемых сообщений. В PGP используются следующие алгоритмы: RSA, SHA, DES, CAST, IDEA, DSS. Закодированная информация сохраняется в виде файла, который может быть передан по сети любым способом (электронная почта, FTP). Для удобства работы PGP может интегрироваться с почтовыми программами, такими как OutlookExpress, TheBat, Eudora. Еще одно достоинство PGP – то, что существуют свободно распространяемые версии этой программы.

Применение программы PGP и подобных ей может оказаться неудобным вследствие того, что пользователю необходимо самому принимать меры для шифрации сообщений. Для того чтобы сделать процедуру шифрования прозрачной для пользователя, существуют различные сетевые протоколы, реализующие технологии защищенных соединений. Протокол SSL (Secure Socket Layer) спроектирован компанией Netscape для своего браузера Netscape Navigator для обеспечения конфиденциальности обмена между двумя прикладными процессами клиента и сервера. Он предоставляет также возможность аутентификации сервера и, опционно, клиента. SSL работает на представительском уровне поверх протокола TCP.

Преимуществом SSL является то, что он независим от прикладного протокола. Прикладные протоколы, такие как HTTP, FTP, TELNET и другие могут работать поверх протокола SSL совершенно прозрачно. Протокол SSL может согласовывать алгоритм шифрования и ключ сессии, а также аутентифицировать сервер до того как приложение примет или передаст первый байт данных. Все протокольные прикладные данные передаются зашифрованными с гарантией конфиденциальности.

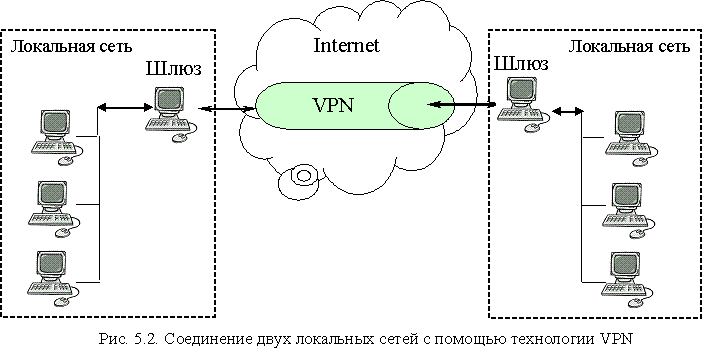
Протокол SSL использует следующие криптографические алгоритмы: DES, DSA, MD5, RC2, RC4. Все современные браузеры поддерживают протокол SSL. Если пользователь вводит в адресной строке URL начинающийся с аббревиатуры HTTPS, то начинает работать протокол HTTPS, который представляет собой стандартный протокол HTTP, защищенный средствами SSL. При этом подключение происходит к порту номер 443, который для HTTPS обычно используется по умолчанию. После этого браузер и сервер обмениваются пакетами, проводя взаимную аутентификацию по имеющимся у них сертификатам, обмениваются сеансовыми ключами шифрования и начинают обмен информацией в зашифрованном виде (рис. 6.1).



Альтернативой использованию протокола SSL может стать применение протокола TLS (Transport Layer Security), имеющего практически аналогичную функциональность.

Для защиты информации на более низком – сетевом – уровне используется технология VPN (Virtual Private Network). VPN (рис.6.2) предполагает использование криптографических методов защиты для обеспечения конфиденциальности и целостности информации на сетевом уровне с использованием ряда современных протоколов сетевого уровня (IPSec, PPTP, L2TP, L2P) при передаче информации по сетям общего пользования. Шифрование происходит прозрачно для пользователей программными или программно-аппаратными средствами. При этом данные всех сетевых пакетов, начиная с транспортного уровня, шифруются и помещаются в закодированном виде в область данных пакета сетевого уровня. Так образуется скрытый от посторонних «туннель», по которому информация может передаваться в сети общего доступа между узлами воображаемой, виртуальной сети. По сравнению с протоколом SSL VPN предоставляет возможность скрыть от злоумышленника такую информацию, как номера используемых портов, а при использовании механизма туннелирования и используемые в локальной сети IP-адреса, что затрудняет возможности сканирования сети, поиска слабых мест в ней.

Стандарт IP Security - это комплект протоколов, касающихся вопросов шифрования, аутентификации и обеспечения защиты при транспортировке IP- пакетов, который должен стать частью разрабатываемого стандарта IPv6. Он включает в себя 3 основных протокола – аутентификации заголовка (Authentification Header, AH), безопасного сокрытия содержимого (Encapsulation Security Payload, ESP) и обмена ключами (Internet Key Exchange, IKE). Протокол AH обеспечивает аутентификацию отправителя пакета и целостность заголовка пакета с использованием механизмов хэширования. Протокол ESP обеспечивает проверку целостности датаграммы пакета и шифрование информации в ней. Он может работать в двух режимах: транспортном и туннельном. В транспортном режиме протокол ESP обеспечивает защиту пакетов (и данных, и заголовков) протоколов более высокого уровня, заголовок пакета IP не защищается. В туннельном режиме ESP обеспечивает связь между двумя шлюзами локальных сетей, защищая информацию в заголовке IP-пакета. В этом режиме IP- пакет полностью кодируется и помещается в область данных пакета, передаваемого между двумя шлюзами. На шлюзе принимающей стороны этот пакет распаковывается и отправляется адресату.



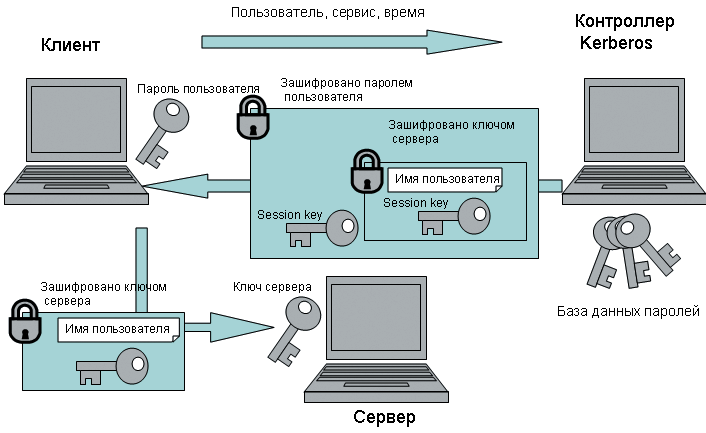
Протокол IKE отвечает за процедуру установки соединения, когда стороны договариваются об используемых криптоалгоритмах, обмениваются сеансовыми ключами шифрования. Протоколы стандарта IPSec используют такие криптоалгоритмы, как DES, TripleDES, AES, SHA, MD5.

Для конечного пользователя простым и дешевым решением является использование программной реализации, можно, например, воспользоваться встроенными средствами поддержки IPSec в ОС Windows.

Решением проблемы регистрации пользователей в локальной сети является криптопротокол Kerberos. Kerberos это система сетевой аутентифиации, основанная на принципах доверия третьей стороне. Другие две стороны - это пользователь и сервис, на котором он хочет авторизоваться. В Kerberos как пользователь, так и сетевая служба не различаются между собой и именуются principal (принципал – юридический термин, означающий лицо, поручающее агенту совершить сделку от его имени), что весьма точно описывает функции принципалов в Kerberos. Принципалы определяются своим именем и паролем, причем в случае сетевой службы в качестве этого пароля выступает ключ, хранящийся на том же компьютере, где работает защищаемый сервис. Kerberos представляет собой трехсторонний процесс, в котором роль посредника выполняет сервис под названием Key Distribution Center (KDC), подтверждающий личность компьютера по запросу другого компьютера и поставляющий ключи для установления между ними защищенного соединения.

Каждый из компьютеров, участвующих в сеансе связи, «делит секрет» с KDC, который состоит из двух компонентов: сервера аутентификации и сервера выдачи мандатов. Если KDC получает запрос о неизвестном ему сервере назначения, он перенаправляет аутентификационную транзакцию на другой KDC, располагающий нужными сведениями.

Обмениваясь с клиентом серией шифрованных сообщений, называемых мандатами, KDC генерирует новые ключи шифрования для каждого этапа процесса аутентификации. Он может успешно подтвердить личность одного компьютера по запросу другого, не выдавая секретных ключей ни одной из сторон и не требуя ни от одной из них постоянного хранения ключей доступа ко всем компьютерам, с которыми они когда-либо установят соединение. Мандат действителен в течение заданного периода времени и может использоваться только одним определенным компьютером для подключения к другому определенному компьютеру (рис. 6.3).

Рис .6.3 Процесс обмена сертификатами в Kerberos

После выдачи мандата клиент может использовать его для получения доступа к целевому серверу неограниченное число раз, но только в период действия мандата. Ни клиент, ни кто-то другой не могут считать или модифицировать мандат, не сделав его недействительным.

KryptoKnight (КриптоРыцарь) является системой проверки подлинности и распределения ключей, разработанной в IBM. Это протокол с секретным ключом, использующий либо DES или модифицированную версию MD5. KryptoKnight поддерживает четыре сервиса безопасности:

— Проверка подлинности пользователя (называемая единственной подписью - single sign-on)

— Двусторонняя проверка подлинности

— Распределение ключей

— Проверка подлинности содержания и происхождения данных

С точки зрения пользователя, KryptoKnight похож на Kerberos. Вот некоторые отличия:

— Для проверки подлинности и шифрования мандатов KryptoKnight использует хэш-функцию.

— KryptoKnight не использует синхронизированных часов, используются только текущие запросы.

— Если Алисе нужно связаться с Бобом, одна из опций KryptoKnight позволяет Алисе послать сообщение Бобу, а затем позволяет Бобу начать протокол обмена ключами.

KryptoKnight, как и Kerberos, использует мандаты и удостоверения. Он содержит и TGS, но в KryptoKnight называются серверами проверки подлинности.