Практическая работа №10

ИССЛЕДОВАНИЕ ПРОТОКОЛОВ УДАЛЕННОЙ АУТЕНТИФИКАЦИИ

Цель работы. Изучить протоколы удаленной аутентификации.

Краткие сведения из теории

1.1 Основные понятия

Идентификация – процедура распознавания субъекта (пользователя, процесса, действующего от имени пользователя, аппаратно-программного компонента) по его уникальному идентификатору, присвоенному субъекту ранее и занесенному в базу данных в момент его регистрации как легального пользователя системы.

Аутентификация – процедура проверки подлинности входящего в систему субъекта, предъявившего свой идентификатор. Аутентификацию не следует путать с авторизацией (процедурой предоставления субъекту определённых прав).

В любой системе аутентификации обычно можно выделить элементы:

– субъект, проходящий процедуру аутентификации;

– характеристика субъекта – отличительная черта;

– хозяин системы аутентификации, несущий ответственность и контролирующий её работу;

– сам механизм аутентификации, то есть принцип работы системы;

– механизм, предоставляющий или лишающий субъекта определенных прав доступа.

Субъект может подтвердить свою подлинность, предъявив, один из следующих аутентификаторов:

1. Аутентификация на основе «субъект знает» –претендент обладает информацией, которой нет у других объектов компьютерной системы (пароль, персональный идентификационный номер, секретный ключ). Эту информацию субъект демонстрирует в протоколах типа «запрос – ответ».

2. Аутентификация на основе «субъект обладает» – претендент имеет некий физический предмет (магнитную карту, интеллектуальную карту, генератор паролей), который выполняет для него криптографические преобразования информации;

3. Аутентификация на основе «субъект есть» – проверяются некоторые биометрические данные человека – голос, радужная оболочка глаза, отпечатки пальцев и др.).

Фактор аутентификации – определенный вид информации, предоставляемый субъектом системе пр его аутентификации. Если в процессе аутентификации используется только один способ аутентификации, то она называется однофакторной аутентификацией, а если несколько – то многофакторной.

Для идентификации средствами криптографии все эти три метода аутентификации могут быть сведены к одному – к аутентификации на основе владения какой-либо информацией.

Действительно, любые биометрические данные или информация, заключенная на физическом носителе, могут бытьпреобразованы в уникальный ключ (при идентификации при помощи криптографической системы или протокола) или пароль (при аутентификации или идентификации парольными схемами), который будет однозначно определять субъекта.

Протокол аутентификации – криптографический протокол, в ходе которого одна сторона убеждается: 1) в идентичности другой стороны, вовлеченной в протокол и 2) в активности другой стороны во время или непосредственно перед моментом приобретения доказательства.

Аутентификация может быть односторонней (когда клиент доказывает свою подлинность серверу) и двусторонней или взаимной (это обоюдная аутентификация между сторонами обмена информацией). Пример односторонней аутентификации – процедура входа в систему (WINDOWS NT). Пример двусторонней – использование протокола KERBEROS (WINDOWS 2000).

Характеристики протоколов аутентификации:

– вычислительная эффективность – количество операций, необходимых для выполнения протокола;

– коммуникационная эффективность – данное свойство отражает количество сообщений и их длину, необходимую для осуществления аутентификации;

– наличие третьей стороны – примером третьей стороны может служить доверенный сервер распределения симметричных ключей или сервер, реализующий дерево сертификатов открытых ключей;

– основа гарантий безопасности – примером могут служить протоколы, обладающие свойством доказательства с нулевым знанием;

– хранения секрета – имеется в виду способ хранения критичной ключевой информации.

Также можно классифицировать протоколы аутентификации по уровню обеспечиваемой безопасности или возможности противостоять определенному классу атак. В соответствии с данным подходом протоколы аутентификации делят на типы:

– простая аутентификации (на основе использования паролей);

– строгая аутентификации (на основе использования криптографических методов и средств);

– протоколы, обладающие свойством доказательства с нулевым знанием.

Общая схема, используемая практически всеми протоколами аутентификации, состоит из следующих действий. Алиса (пользователь А) желает установить защищенное соединение с Бобом (пользователь В) или считающимся надежным **Центром распространения ключей** (**KDC** – **Key Distribution Center**). Затем в разных направлениях посылаются еще несколько сообщений. По мере их передачи нарушитель (Труди) может перехватить, изменить и снова воспроизвести эти сообщения, чтобы обмануть Алису и Боба или просто сорвать передачу информации.

Тем не менее, когда протокол завершит свою работу, Алиса должна быть уверена, что разговаривает с Бобом, а Боб – что разговаривает с Алисой. Кроме того, в большинстве протоколов собеседники также установят секретный **ключ сеанса** (**session key**), которым будут пользоваться для последующего обмена информацией. На практике весь обмен данными шифруется с помощью одного из алгоритмов с секретным ключом (AES или тройной DES), так как их производительность намного выше производительности алгоритмов с открытым ключом. Тем не менее алгоритмы с открытым ключом широко применяются в протоколах аутентификации и для определения ключа сеанса.

Цель использования нового, случайно выбираемого ключа сеанса для каждого нового соединения состоит в минимизации трафика, посылаемого с использованием закрытых и открытых ключей пользователя, уменьшении количества шифрованного текста, который может достаться злоумышленнику, а также минимизации вреда, причиняемого в случае, если процесс даст сбой. Поэтому после установления соединения в процессе должен храниться только один временный ключ сеанса. Все постоянные ключи должны быть тщательно стерты.

1.2 Аутентификация, основанная на общем секретном ключе

Предположим, что у Алисы и Боба есть общий секретный ключ K_{AB}. Об этом секретном ключе можно договориться при личной встрече или по телефону, но в любом случае не по (незащищенной) сети.

В основе этого протокола лежит принцип, применяемый во многих протоколах аутентификации: одна сторона посылает другой случайное число, которое другая сторона преобразует особым образом и возвращает результат. Такие протоколы называются протоколами типа **оклик-отзыв** (**challenge-response**). В этом и последующих протоколах аутентификации будут использоваться следующие условные обозначения:

– A и B – Алиса и Боб;
– R_i – оклик, где индекс означает его отправителя;
– K_i – ключи, где индекс означает владельца ключа;
– K_S – ключ сеанса.

Последовательность сообщений протокола аутентификации с общим ключом показана на рисунке 1. В первом сообщении Алиса посылает свое удостоверение личности (A) Бобу тем способом, который ему понятен. Боб, конечно, не знает, пришло ли это сообщение от Алисы или от злоумышленника, поэтому он выбирает большое случайное число R_B и посылает его в качестве оклика «Алисе» открытым текстом (сообщение 2). Затем Алиса шифрует это сообщение секретным ключом, общим для нее и Боба, и отправляет шифрованный текст $K_{AB}(R_B)$ в сообщении 3. Когда Боб видит это сообщение, он понимает, что оно пришло от Алисы, так как злоумышленник не должен знать ключа K_{AB}, и поэтому он не смог бы сформировать такое сообщение.

Более того, поскольку оклик R_B выбирался случайно в большом пространстве чисел (например, 128-битных случайных чисел), очень маловероятно, чтобы злоумышленник мог уже видеть этот оклик и ответ на него в предыдущих сеансах.

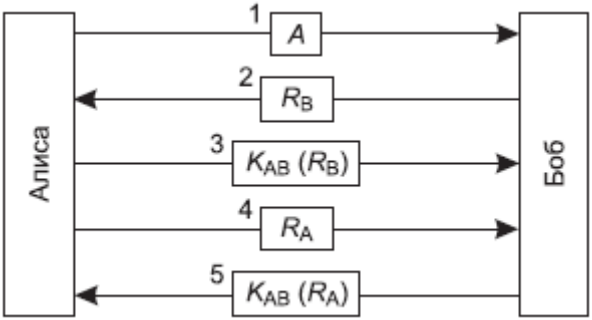

Рисунок 1 – Двусторонняя аутентификация при помощи протокола оклик-отзыв

К этому моменту Боб уверен, что говорит с Алисой, однако Алиса еще пока не уверена ни в чем. Злоумышленник мог перехватить сообщение 1 и послать обратно оклик R_B. Далее протокол работает симметрично: Алиса посылает оклик, а Боб отвечает на него. Теперь уже обе стороны уверены, что говорят именно с тем, с кем собирались. После этого они могут установить временный ключ сеанса KS, который можно переслать друг другу, закодировав его все тем же общим ключом K_{AB}.

Количество сообщений в этом протоколе можно сократить, объединив в каждом сообщении ответ на предыдущее сообщение с новым окликом, как показано на рисунке 2.

Здесь Алиса сама в первом же сообщении посылает Бобу оклик. Отвечая на него, Боб помещает в то же сообщение свой оклик. Таким образом, вместо пяти сообщений понадобилось всего три.

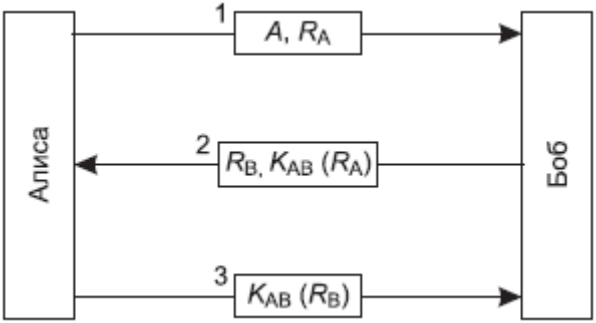

Рисунок 2 – Укороченный двусторонний протокол аутентификации

При некоторых обстоятельствах злоумышленник может атаковать этот протокол способом, известным под названием **зеркальная атака** (**reflection attack**). В частности, Труди может взломать его, если ей будет позволено одновременно открыть несколько сеансов связи с Бобом.

Схема зеркальной атаки показана на рисунке 3. Она начинается с того, что Труди, объявляя себя Алисой, посылает оклик R_T. Боб, как обычно, отвечает своим собственным окликом R_B.

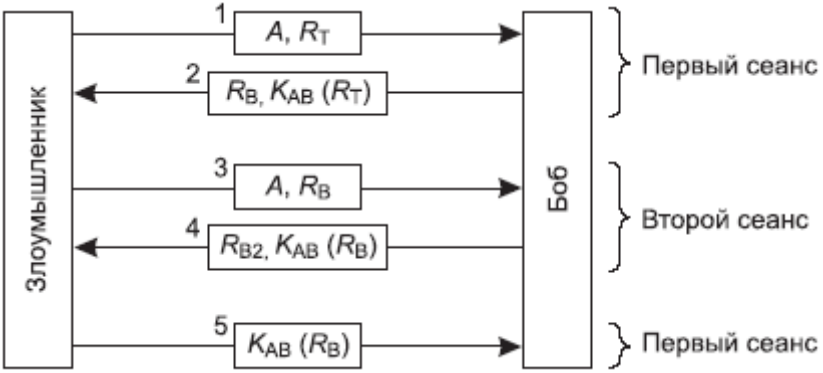

Рисунок 3 – Зеркальная атака

Злоумышленник может открыть второй сеанс сообщением 3 и подать в качестве оклика Бобу оклик самого Боба, взятый из второго сообщения. Боб спокойно шифрует его и посылает обратно $K_{AB}(R_B)$ в сообщении 4. Теперь у Труди есть необходимая информация, поэтому она завершает первый сеанс и прерывает второй. Боб теперь уверен, что злоумышленник – это Алиса.

Приведем четыре общих правила организации удаленной аутентификации:

1. Инициатор сеанса должен подтверждать свою личность прежде, чем это сделает отвечающая сторона. Это помешает злоумышленнику получить ценную для него информацию, прежде чем он подтвердит свою личность.

2. Следует использовать два раздельных общих секретных ключа: один для инициатора сеанса, а другой для отвечающего, K_{AB} и K'_{AB}.

3. Инициатор и отвечающий должны выбирать оклики из различных непересекающихся наборов. Например, инициатор должен пользоваться четными номерами, а отвечающий – нечетными.

4. Протокол должен уметь противостоять атакам, при которых запускается второй параллельный сеанс, информация для которого извлекается при помощи первого сеанса (или наоборот).

Если нарушается хотя бы одно из этих правил, протокол оказывается уязвимым.

1.3 Аутентификация с помощью центра распространения ключей

Другой подход состоит в организации доверительного центра распространения ключей (**KDC, key distribution center**). При такой схеме у каждого пользователя всего один ключ, общий с KDC. Операции с ключами аутентификации и сеансовыми ключами проходят через KDC. Простейший протокол аутентификации с помощью центра распространения ключей, включающий две стороны и доверенный KDC, изображен на рисунке 4.

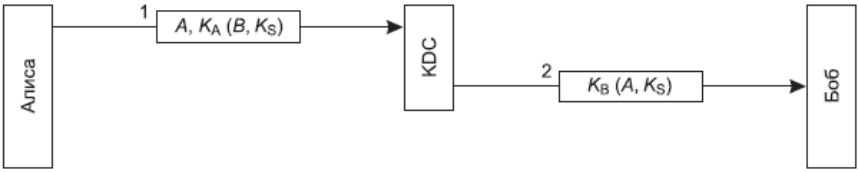

Рисунок 4 – Первая попытка протокола аутентификации с помощью KDC-центра

Идея, лежащая в основе протокола, проста: Алиса выбирает ключ сеанса, K_S, и заявляет KDC, что она желает поговорить с Бобом при помощи ключа K_S. Это сообщение шифруется секретным ключом K_A, которым совместно владеют только Алиса и центр распространения ключей. Центр распространения ключей расшифровывает это сообщение и извлекает из него иденти-

фикатор личности Боба и ключ сеанса. Затем он формирует новое сообщение, содержащее идентификатор личности Алисы и ключ сеанса, и посылает его Бобу. Это сообщение зашифровывается ключом K_B – секретным ключом, общим для Боба и центра распространения ключей. Расшифровав это сообщение, Боб узнает, что Алиса желает с ним поговорить и какой ключ она хочет использовать.

Аутентификация в данном случае происходит сама собой. KDC знает, что сообщение 1 пришло от Алисы, так как больше никто не может зашифровать его секретным ключом Алисы. Аналогично, Боб знает, что сообщение 2 пришло от KDC, так как кроме него их общий секретный ключ никому не известен.

К сожалению, этот протокол содержит серьезную ошибку – он уязвим к **атаке повторным воспроизведением (replay attack).**

Существует несколько решений этой проблемы. Первое решение состоит в помещении в каждое сообщение временного штампа. Все устаревшие сообщения просто игнорируются.

Труди может обмануть протокол, послав повторное сообщение во время этого интервала.

Второе решение заключается в помещении в сообщение уникального номера, обычно называемого **нонсом** (**nonce**). Каждая сторона должна запоминать все предыдущие нонсы и отвергать любое сообщение, содержащее использованный ранее нонс. Однако нонсы должны храниться вечно, иначе Труди попытается воспроизвести сообщение пятилетней давности. Кроме того, если машина потеряет список нонсов в результате сбоя, она снова станет уязвимой к атакам повторным воспроизведением. Можно комбинировать временные штампы и нонсы, чтобы ограничить срок хранения нонсов, но так или иначе, протокол должен быть значительно усложнен.

Более сложный метод аутентификации состоит в использовании многостороннего протокола оклик-отзыв. Хорошо известным примером такого протокола является **протокол аутентификации Нидхэма-Шредера (Needham-Schroeder authentication protocol**), один из вариантов которого показан на рисунке 5.

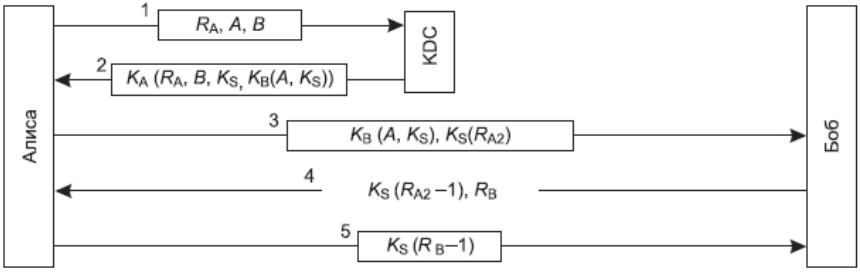

Рисунок 5 – Протокол аутентификации Нидхэма-Шредера

Работа протокола начинается с того, что Алиса сообщает KDC, что она желает поговорить с Бобом. Это сообщение содержит в качестве нонса большое случайное число R_A. Центр распространения ключей посылает обратно сообщение 2, содержащее случайное число Алисы, ключ сеанса и так называемый билет, который она может послать Бобу. Цель посылки случайного числа R_A состоит в том, чтобы убедить Алису, что сообщение 2 является свежим, а не повторно воспроизведенным. Идентификатор Боба также помещается в сообщение 2, на случай, если злоумышленник (Труди) вздумает заменить его идентификатор на свой в сообщении 1, так чтобы KDC зашифровал билет в конце сообщения 2 ключом K_T (ключ Труди), вместо K_B. Билет, зашифро ванный ключом K_B, помещается внутри зашифрованного сообщения, чтобы злоумышленник не смог заменить его чем-либо другим, пока сообщение 2 добирается до Алисы.

Затем Алиса посылает билет Бобу вместе с новым случайным числом R_{A2}, зашифрованным ключом сеанса K_S. В сообщении 4 Боб посылает обратно $K_S(R_{A2} - 1)$, чтобы доказать Алисе, что она разговаривает с настоящим Бобом. Отсылать обратно просто $K_S(R_{A2})$ бессмысленно, так как это число могло быть украдено злоумышленником из сообщения 3.

Получив сообщение 4, Алиса убеждается, что разговаривает с Бобом и что до сих пор не было использовано повторных сообщений. Между отправкой случайного числа R_{A2} и получением ответа на него в виде $K_S(R_{A2} - 1)$ проходит довольно короткий промежуток времени. Цель сообщения 5 – убедить Боба, что он действительно разговаривает с Алисой, и что в этом сеансе связи также отсутствуют повторно воспроизведенные данные. Возможность атаки с помощью повторного воспроизведения ранее записанной информации исключается этим протоколом благодаря тому, что каждая сторона формирует оклик другой стороны и получает на него отзыв.

Несмотря на всю кажущуюся солидность протокола, в нем, тем не менее, имеется небольшое слабое место. Если злоумышленнику удастся каким-либо способом раздобыть старый ключ сеанса K_S, он сможет инициировать новый сеанс с Бобом, повторно воспроизведя сообщение 3 с использованием скомпрометированного ключа, и выдать себя за Алису.

На рисунке 6 показан слегка видоизмененный протокол Отуэя-Риса, который решает эту проблему.

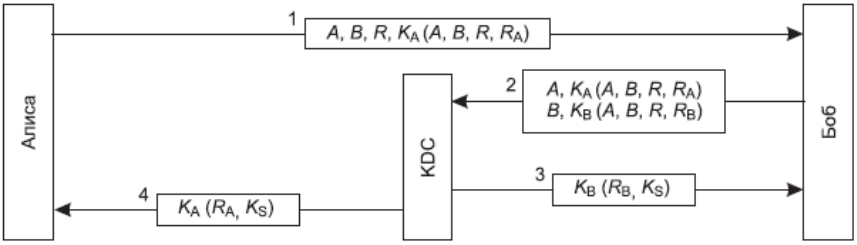

Рисунок 6 – Протокол аутентификации Отуэя-Риса

В протоколе Отуэя-Риса Алиса начинает с формирования пары случайных номеров: R, который будет использоваться в качестве общего идентификатора, и R_A, который Алиса будет использовать в качестве оклика Боба. Получив это сообщение, Боб формирует новое сообщение из зашифрованной части сообщения Алисы и аналогичной собственной части. Обе части сообщения, зашифрованные ключами K_A и K_B, идентифицируют Алису и Боба, содержат общий идентификатор и оклики.

Центр распространения ключей проверяет, совпадают ли общие идентификаторы R в обеих частях сообщения. Они могут не совпадать, если злоумышленник подменил R в сообщении 1 или заменил часть сообщения 2. Если оба общих идентификатора R совпадают, KDC считает сообщение, полученное от Боба, достоверным. Затем он формирует ключ сеанса K_S и отправляет его Алисе и Бобу, зашифровав ключ сеанса ключами Алисы и Боба. Каждое сообщение также содержит случайное число получателя, в доказательство того, что эти сообщения посланы KDC, а не злоумышленником.

К этому моменту Алиса и Боб обладают одним и тем же ключом сеанса и могут начать обмен информацией. После первого же обмена данными они увидят, что обладают одинаковыми копиями ключа сеанса K_S, на чем процесс аутентификации можно будет считать завершенным.

1.4 Аутентификация при помощи протокола Kerberos

В операционной системе Windows 2000 и более поздних версиях применяется протокол аутентификации **Kerberos**, основанный на одном из вариантов протокола Нидхэма-Шредера. Он назван по имени трехглавого пса греческих мифов Цербера, охранявшего выход из Аида.

Протокол Kerberos был разработан в Массачусетском технологическом институте для обеспечения пользователям рабочих станций надежного доступа к сетевым ресурсам. Его основное отличие от протокола Нидхэма-Шредера состоит в предположении о довольно хорошей синхронизации всех часов в сети. Было разработано несколько последовательных версий

протокола. Версия V5 наиболее широко применяется в промышленности и описана в RFC 4120.

В работе протокола Kerberos, помимо рабочей (клиентской) станции Алисы, принимают участие еще три сервера:

1. **Сервер аутентификации** (**AS**, **Authentication Server**) – проверяет личность пользователей при входе в сеть.

2. **Сервер выдачи билетов** (**TGS**, **Ticket Granting Server**) – выдает «билеты, подтверждающие подлинность».

3. Боб, то есть сервер, предоставляющий услуги Алисе.

Сервер аутентификации AS аналогичен центру распространения ключей KDC в том, что у него есть общий секретный пароль для каждого пользователя. Работа сервера выдачи билетов TGS состоит в выдаче свидетельств, убеждающих другие серверы в том, что владелец билета действительно является тем, за кого он себя выдает.

Чтобы начать сеанс, Алиса вводит свое имя и название TGS. Рабочая станция посылает введенное имя открытым текстом на сервер аутентификации, как показано в сообщении 1 (рисунок 7). Сервер аутентификации AS возвращает рабочей станции

Алисы ключ сеанса и билет $KTGS(A, KS, t)$ для сервера выдачи билетов TGS. Ключ сеанса зашифровывается секретным ключом Алисы, так чтобы только Алиса могла его расшифровать. Только после получения сообщения 2 рабочая станция запрашивает пароль Алисы, и никак не раньше. С помощью этого пароля формируется ключ K_A, которым расшифровывается сообщение 2 и из него извлекается ключ сеанса. После расшифровки рабочая станция сразу же уничтожает хранящийся в ее памяти пароль.

Если вместо Алисы на рабочей станции попытается зарегистрироваться Труди, введенный ею пароль окажется неверным, что будет обнаружено рабочей станцией, так как стандартная часть сообщения 2 окажется неверной.

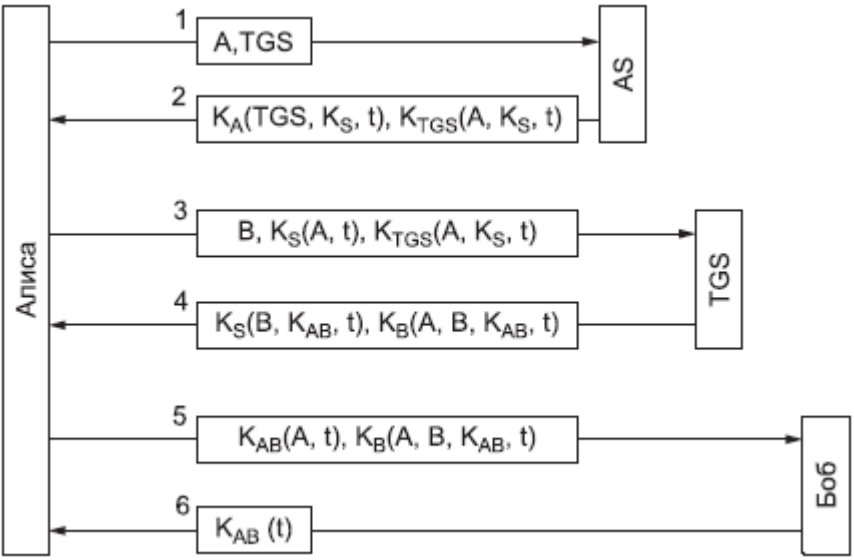

Рисунок 7 – Работа протокола Kerberos V5

После регистрации в сети Алиса может сообщить рабочей станции, что она хочет вступить в контакт с файловым сервером, то есть Бобом. При этом рабочая станция посылает серверу выдачи билетов сообщение 3 с просьбой выдать билет для общения с Бобом. Ключевым элементом этого запроса является билет $K_{TGS}(A, K_S, t)$, который зашифрован секретным ключом TGS-сервера и используется для подтверждения личности отправителя. Сервер выдачи билетов отвечает созданием ключа сеанса K_{AB}, которым будут пользоваться Алиса и Боб. Он отправляет Алисе две версии этого ключа. Один ключ зашифрован ключом сеанса K_S, поэтому Алиса может его прочитать. Второй ключ представляет собой еще один билет, который шифруется ключом Боба K_B, что позволяет Бобу его прочитать.

Злоумышленник может скопировать сообщение 3 и попытаться использовать его снова, но ему помешает временной штамп t, отправляемый вместе с этим сообщением.

Злоумышленник не может заменить этот временной штамп на более новый, так как не знает ключа сеанса K_S, которым пользуется Алиса для общения с сервером выдачи билетов. Даже если злоумышленник очень быстро повторит сообщение 3, все равно, единственное, что он получит в ответ, это сообщение 4, которое он не смог расшифровать в первый раз и не сможет расшифровать и во второй раз.

После этого Алиса через новый билет может послать Бобу ключ K_{AB} для установки сеанса с Бобом. Эти сообщения также содержат временные штампы. Сообщение 6, получаемое в ответ (в качестве возможной проверки), подтверждает, что Алиса говорит именно с Бобом, а не со злоумышленником.

Наконец, после этой серии обмена сообщениями Алиса сможет обмениваться с Бобом данными, используя ключ сеанса K_{AB}. Если после этого Алиса решит, что ей необходим другой сервер, например Кэрол (Carol, C), она может просто послать серверу выдачи билетов сообщение, аналогичное третьему, заменив в нем *B* на *C* (то есть идентификатор Боба на идентификатор Кэрол). TGS мгновенно ответит сообщением, содержащим билет, зашифрованный ключом K_C. Этот билет Алиса пошлет Кэрол, для которой он будет служить гарантией подлинности Алисы.

Достоинство этого протокола состоит в том, что теперь Алиса может получать защищенный доступ к любому серверу сети, и в то же время ее пароль ни разу не передавался по сети. В действительности, он только на несколько миллисекунд появлялся в ее рабочей станции. Каждый сервер выполняет свою собственную процедуру авторизации. Когда Алиса предъявляет свой билет Бобу, это всего лишь подтверждает Бобу подлинность предъявителя билета. К чему же Алиса может получить доступ на сервере, решает Боб.

1.5 Аутентификация с помощью шифрования с открытым ключом

Взаимная аутентификация также может выполняться с помощью шифрования с открытым ключом. Для начала Алисе нужно получить открытый ключ Боба. Если инфраструктура **PKI** (**Public Key Infrastructure** – **инфраструктура систем с открытыми ключами**) реализована на основе сервера каталогов, выдающего сертификаты на открытые ключи, Алиса может потребовать сертификат Боба, что показано в виде сообщения 1 на рисунке 8. Ответ, содержащийся в сообщении 2, – это сертификат X.509 с открытым ключом Боба. Проверив корректность подписи, Алиса может отправить Бобу сообщение со своим идентификатором и нонсом.

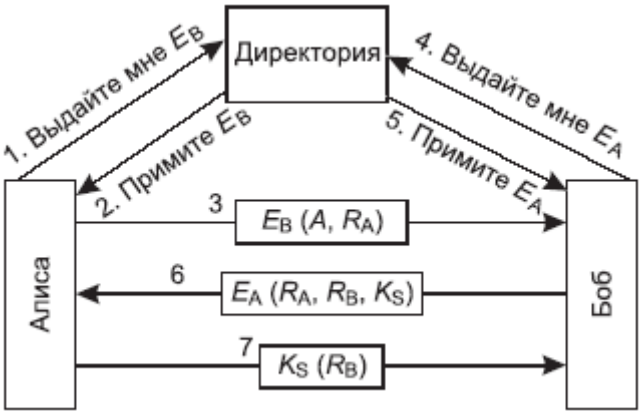

Рисунок 8 – Взаимная идентификация с помощью открытого ключа

Когда Боб получает это сообщение, он не знает, пришло ли оно от Алисы или от Труди (злоумышленника), но он делает вид, что все в порядке и просит сервер каталогов выдать ему открытый ключ Алисы (сообщение 4). Вскоре он его получает (в сообщении 5). Затем он отсылает Алисе сообщение 6, содержащее случайное число Алисы R_A, свой собственный нонс R_B и предлагаемый ключ сеанса K_S. Алиса расшифровывает полученное сообщение 6 своим закрытым ключом. Она видит в нем свое случайное число R_A и очень этому рада: это подтверждает, что сообщение пришло от Боба, так как у Труди не должно быть способа определить значение этого числа. Кроме того, случайное число R_A свидетельствует о свежести этого сообщения. Алиса соглашается на установку сеанса, отправляя сообщение 7. Когда Боб видит свое случайное число R_B, зашифрованное ключом сеанса, который он сам же сформировал, он понимает, что Алиса получила сообщение 6 и проверила значение R_A. Боб счастлив и доволен.

Злоумышленник может сфабриковать сообщение 3 и спровоцировать Боба на проверку Алисы, но Алиса увидит число R_A, которого она не посылала, и не станет продолжать. Злоумышленник не сможет убедительно подделать сообщение 7, так как ему не известны значения оклика R_B или ключа K_S, и он не может определить их, не имея закрытого ключа Алисы.

Порядок выполнения работы

1 Изучить краткие сведения из теории.

2 Нарисовать алгоритм зеркальной атаки двусторонней аутентификации при помощи протокола оклик-отзыв (рисунок 1).

3 Воспроизвезти по этапам алгоритмы аутентификации Нидхэма-Шредера и Отуэя-Риса, задавая имена пользователей, сеансовые ключи и нонсы.

4 Воспроизвезти по этапам алгоритм аутентификации Kerberos V5, задавая имена пользователей, сеансовые ключи и нонсы.

Содержание отчета

1 Цель работы.

2 Алгоритм зеркальной атаки двусторонней аутентификации при помощи протокола оклик-отзыв.

3 Иллюстрации этапов алгоритмов аутентификации Нидхэма-Шредера, Отуэя-Риса, Kerberos V5.

4 Вывод по работе.

Контрольные вопросы

1 Что такое идентификация?
2 Что такое аутентификация?
3 Что такое протокол аутентификации?
4 Протоколы аутентификации с секретным ключом.
5 Протокол аутентификации Kerberos V5.
6 Протоколы аутентификации с открытым ключом