5. http://www.leader.ru/secure/who.html.

6. http://www.privasec.com.

Статья поступила в редакцию 25.03.2004

Владислав Андреевич Киселенко родился в 1962 г., окончил в 1983 г. Московский военный институт. Преподаватель Академии военных наук. Автор 10 научных работ в области компьютерной безопасности.

V.A. Kiselenko (b. 1962) graduated from the Moscow Military Institute in 1983. Teacher of Academy of Military Sciences. Author of 10 publications in the field of computer safety.

УДК 681.3

Н. В. Медведев

КОНЦЕПЦИЯ ОТКРЫТЫХ СИСТЕМ И ЗАДАЧА АУТЕНТИФИКАЦИИ В ЛОКАЛЬНЫХ СЕТЯХ

Рассмотрена задача обеспечения безопасности локальных компьютерных сетей, решаемая с помощью специализированных программных средств — протоколов сетевой аутентификации. Показаны возможности таких протоколов для подтверждения полномочий пользователей сети при организации доступа к информационным ресурсам на примере протокола сетевой аутентификации Kerberos.

Протокол сетевой аутентификации. За последние десятилетия значительно возросла роль компьютерных вычислительных систем и сетей. С их помощью обрабатываются огромные объемы различной информации. Также в последнее время многократно увеличилась степень открытости компьютерных сетей, их размеры и уровень их взаимозависимости, появились распределенные системы обработки информации колоссального размера. В связи с этим возникает необходимость разграничения доступа к обрабатываемой, передаваемой и хранимой информации с учетом специфики систем. Для решения задач этого типа был разработан протокол сетевой аутентификации Kerberos. В настоящей работе рассматриваются основные способы функционирования данного протокола, а также его расширений в операционной системе MS Windows 2000.

Протокол Kerberos является протоколом сетевой аутентификации, с помощью которого решается следующая задача: в открытой, незащищенной сети, в узлах которой находятся некоторые субъекты — пользователи, необходимо обеспечить их взаимную аутентификацию (т.е. подтверждение их полномочий). Протокол аутентификации Kerberos

для клиента и сервера ресурсов является третьей доверенной стороной, реализующей попарную проверку подлинности. Субъектами с точки зрения системы Kerberos могут являться пользователи, компьютеры, программы-клиенты и программы-серверы. Каждый субъект обладает своим секретным ключом, и для того, чтобы субъект-клиент получил доступ к ресурсам субъекта-сервера, он должен продемонстрировать субъекту-серверу знание своего секретного ключа. Необходимо пояснить, что фактически сервер Kerberos не занимается аутентификацией клиента, он лишь выдает некоторую зашифрованную информацию клиенту, при этом только подлинный клиент может расшифровать полученное сообщение и корректно обработать полученную информацию. Тем самым, возможна правильная аутентификация подлинного пользователя сервером ресурса, а также минимизируется вероятность компрометации пароля. Клиент считается аутентифицированным, если он продемонстрировал знание секретного ключа. Однако возможна, например, пересылка ключа между субъектами, вследствие чего необходим иной способ аутентификации.

Рассмотрим взаимодействие некоторых пользователей A, B и T при использовании протокола Kerberos. Протокол Kerberos представляет собой разновидность протокола Нидхема–Шредера. В рамках данного протокола пользователь A (клиент) и пользователь B (сервер) имеют общие (разделяемые) секретные ключи с пользователем T, их задача — генерировать сеансовый ключ, необходимый для обмена сообщениями. Рассмотрим алгоритм генерации таких ключей.

1. Пользователь A отправляет запрос пользователю T со своим именем и именем пользователя B:

$$(A, B).$$

2. Пользователь T генерирует сообщение, включающее метку времени, значение времени жизни L, случайный сеансовый ключ K и имя пользователя A. Это сообщение пользователь T шифрует с помощью ключа, общего для него и пользователя B. После этого пользователь T генерирует второе сообщение, содержащие метку времени, значение времени жизни L, сеансовый ключ K, имя пользователя B и шифрует его с помощью ключа, общего для пользователей T и A. Оба зашифрованных сообщения пользователь T посылает пользователю A:

$$E_A(T, L, K, B), \quad E_B(T, L, K, A).$$

3. Пользователь A генерирует сообщение, состоящее из его имени и метки времени, шифрует сообщение с помощью сеансового ключа K и посылает пользователю B. Кроме того, пользователь A отправляет

пользователю B сообщение от пользователя T, зашифрованное с помощью ключа пользователя B:

$$E_K(A, T), \quad E_B(T, L, K, A).$$

Таким образом клиент доказывает свою подлинность серверу, получает в свое распоряжение сеансовый (сессионный) ключ, который он может использовать для криптографической защиты сообщений, передаваемых между клиентами.

4. В рамках данного протокола возможно продолжение обмена служебными сообщениями в случае, если необходима взаимная аутентификация клиентов.

Особенностью данного протокола является жесткая синхронизация времени всех пользователей (клиентов и серверов) со временем пользователя T, которая достигается синхронизацией с надежным сервером времени с точностью до нескольких минут.

Функциональные особенности протокола Kerberos. Функциональная схема протокола Kerberos относительно проста. Она подразумевает наличие следующих компонентов (рис. 1):

— сервера начальной аутентификации;

— сервера выдачи билетов;

— сервера приложений, поддерживающего протокол Kerberos;

— клиента, поддерживающего протокол Kerberos;

— удаленного сервера администрирования (который не является обязательным и присутствует не во всех реализациях).

На рис. 1 отмечен порядок передачи сообщений (1–6).

Протокол сетевой аутентификации Kerberos представляет собой набор субпротоколов. Протоколы, используемые системой Kerberos, представлены на рис. 2.

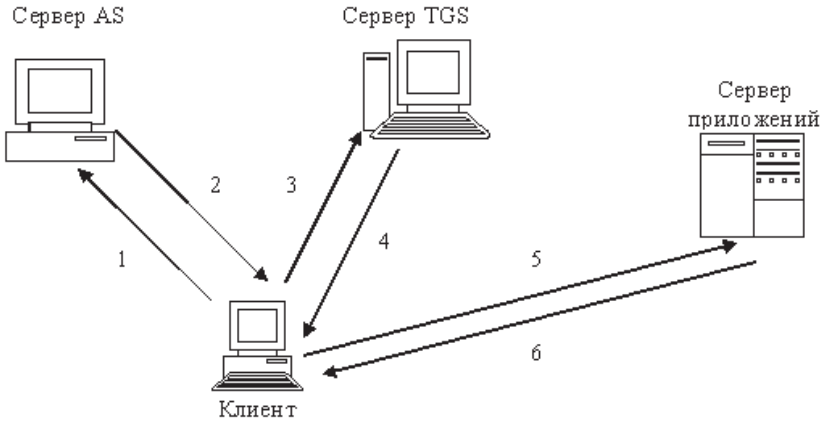

Рис. 1. Функциональная схема протокола Kerberos

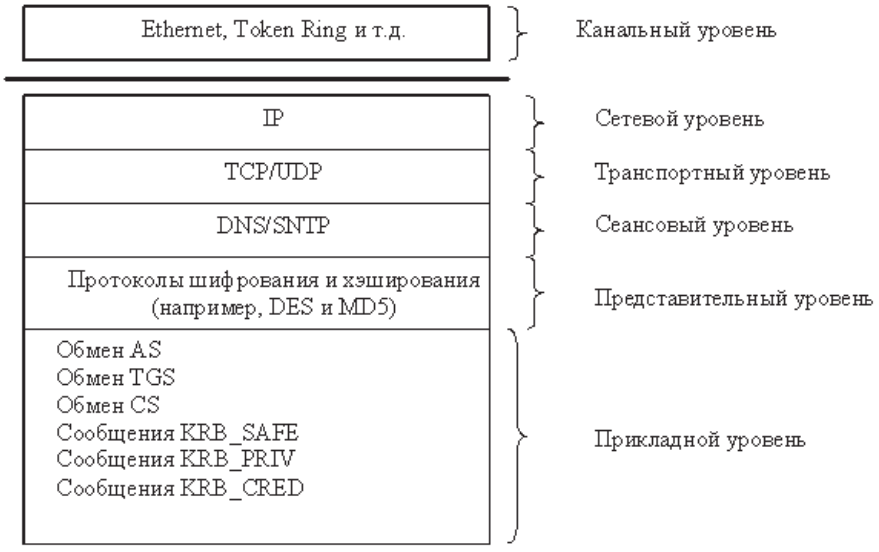

Рис. 2. Схема протоколов в системе Kerberos

Из рис. 2 видно, что на сетевом уровне используется протокол IP вследствие его широкого распространения и приемлемой функциональности. К основным недостаткам протокола IP можно отнести отсутствие криптографической защиты содержимого его пакета и отсутствие аутентификации на сетевом уровне. Последнее особенно важно, так как традиционно предполагается использование IP-адреса для идентификации узла [1]. Применение подобного способа идентификации делает возможным подмену IP-адресов узлов, а следовательно, возможен срыв процесса аутентификации.

На транспортном уровне могут использоваться протоколы стека TCP/IP. В работе [2] рекомендуется использование протокола UDP, однако его использование сопряжено с некоторыми трудностями при аутентификации. Вследствие ненадежности протокола UDP может возникать повторная передача уже принятого другой стороной сообщения. С учетом реализованной в протоколе защиты от воспроизведения (replay) ключа возникновение такой ситуации будет расценено как попытка атаки путем воспроизведения ранее записанного из сети запроса. Результатом этого может стать полный хаос в системе. Также в отдельных реализациях протокола Kerberos размер некоторого сообщения может превысить размер датаграммы протокола IP, что потребует фрагментации сообщения, вследствие чего могут возникнуть дополнительные сложности при его обработке. Поэтому производители программных реализаций протокола Kerberos, в частности компания Microsoft, предпочитают использовать надежный протокол TCP. Заметим, что в некоторых расширениях протокола Kerberos в операционной системе

Windows 2000 возможно возникновение ситуации, при которой размер запроса (1500 байт) будет превышать размер датаграммы протокола UDP; в этих случаях применяется протокол TCP. Однако расширения, сделанные в протоколе Kerberos разработчиками компании Microsoft, специфичны для продуктов именно этой компании. Вследствие этого при посылке запросов (ответов) к серверам (клиентам) на базе иных операционных систем Windows используется протокол UDP, так как нет необходимости в передаче дополнительной информации.

При рассмотрении основных функциональных элементов системы Kerberos нельзя не упомянуть о службе DNS. В современных операционных системах применяется служба DNS с поддержкой возможностей записей поиска служб (SRV). Эти записи необходимы для поиска служб, требующихся клиенту. В данном случае записи SRV необходимы для поиска сервера Kerberos. Клиент обращается к службе DNS с запросом о записях SRV, используя формат Kerberos.udp.DnsDomainName для службы KDC (Key Direct Control — служба управления ключом), в результате чего получает IP-адрес сервера Kerberos. Также IP-адрес сервера Kerberos в некоторых реализациях можно получить из текстового конфигурационного файла.

Ранее упоминалось, что запросы клиента и ответы сервера содержат метку времени для противодействия атакам путем воспроизведения, при которых сообщение, записанное некоторое время назад, повторно воспроизводится. Однако наличие метки времени требует жесткой синхронизации времени всех участников обмена. Именно для синхронизации времени в элементах сети, на которых установлен протокол Kerberos (т.е. в некотором округе), могут использоваться сетевые протоколы времени, например SNTP. К сожалению, сетевые протоколы синхронизации времени, как правило, плохо защищены, что также ведет к снижению информационной безопасности при использовании протокола Kerberos.

Важную роль в функционировании протокола Kerberos играют применяемые в нем алгоритмы шифрования и хэширования. В работе [3] сформулированы требования к криптографическим алгоритмам, которые можно применять в системе Kerberos. Эти алгоритмы должны:

— обеспечивать требуемый уровень криптостойкости шифрованного текста;

— отслеживать случайные и целенаправленные искажения сообщений.

В реализации протокола Kerberos, созданной компанией Microsoft для операционной системы Windows 2000, используется алгоритм шифрования DES и хэш-функция MD5. Возможно применение и других алгоритмов при условии выполнения приведенных требований.

Рассмотрим назначение каждого компонента системы.

Сервер Kerberos. Этот сервер включает в себя следующие элементы:

— сервер начальной аутентификации (сервер AS), в обязанности которого входит выдача билетов на билеты (TGT — Ticket Granting Ticket), проведение предаутентификации и т.д.;

— сервер выдачи билетов (TGS — Ticket Granting Server), который производит выдачу клиентам билетов доступа к конкретным серверам ресурсов;

— базу данных сервера Kerberos, которая содержит информацию, необходимую для корректного управления секретными ключами пользователей. База данных должна включать в себя записи [1] идентификатора принципала, секретного ключа принципала, версии ключа принципала, максимального срока действия билетов, общего максимального срока действия обновляемых билетов. База данных может располагаться на некотором удаленном сервере, а не на самом сервере Kerberos, хотя такой подход не рекомендован в работе [4].

Клиенты системы Kerberos. Клиентами этой системы могут являться персональные компьютеры пользователей, программы-серверы, программы-клиенты и т.д. В этой системе клиент — это некоторый субъект, нуждающийся в услуге, оказываемой протоколом Kerberos. Клиент должен быть способен составить корректный запрос и правильно обработать ответ. Существуют некоторые требования по безопасности клиента, которые будут рассмотрены далее.

Сервер приложений. Этот сервер предоставляет потребителям некоторый сервис и поддерживает протокол сетевой аутентификации Kerberos.

Субпротоколы Kerberos. Субпротоколы Kerberos обеспечивают обмен ключом с сервером начальной аутентификации (Authentification Server Exchange — AS), с сервером выдачи билетов (Ticket Granting Server — TGS), а также аутентификационный обмен типа клиент–сервер (Client Server Exchange — CS). Данный обмен инициируется клиентом в том случае, если клиент желает получить аутентификационный допуск (credentials) к некоторому серверу, но не имеет начального допуска. При этом обмене секретный ключ клиента используется для шифрования и расшифровывания сообщений. Этот обмен возникает, в частности, в следующих случаях:

— при входе пользователя в систему;

— при получении допуска к серверу TGS, который будет позднее использован для получения допуска к другим серверам (без использования секретного ключа пользователя);

— при получении допуска к службам, которым требуется секретный пароль пользователя (например, к серверу смены пароля). Типы сообщений AS-обмена и пути их прохождения представлены в табл. 1.

Таблица 1

Сообщения AS-обмена

Путь прохождения сообщения	Тип сообщения
От клиента к протоколу Kerberos	KRB_AS_REQ
От протокола Kerberos к клиенту	KRB_AS_REP KRB_ERROR

Данный обмен не обеспечивает аутентификации клиента. Допуск, полученный в результате AS-обмена, должен быть использован в TGS-обмене. В качестве запроса клиент посылает в открытой форме [5] сообщение KRB_AS_REQ, содержащее его имя и имя сервера, к которому он получает допуск, а также ряд дополнительных опций, определяющих способы аутентификации.

При получении сообщения типа KRB_AS_REQ сервер AS осуществляет поиск информации о клиенте в своей базе данных и при положительном результате приступает к созданию сообщения типа KRB_AS_REP, содержащего:

— билет, зашифрованный секретным ключом сервера (этот билет клиент должен предъявить серверу);

— сессионный ключ, разделяемый сервером и клиентом и зашифрованный секретным ключом клиента;

— дополнительную информацию для исключения возможности повторного использования (replay) конкретного билета и для связывания ответа KRB_AS_REP с запросом KRB_AS_REQ; дополнительная информация также шифруется секретным ключом клиента.

При получении ответа сервера клиент расшифровывает часть сообщения, содержащую сессионный ключ. Это не представляет сложности, так как секретный ключ клиента является однонаправленной хэш-функцией пароля пользователя. После получения сеансового ключа клиент уничтожает пароль и его хэш-значение. Это необходимо для снижения вероятности компрометации секретного ключа клиента. Даже в случае получения злоумышленником копии памяти клиента будут раскрыты только билет и сеансовый ключ, что представляет значительно меньшую ценность вследствие ограниченности срока их действия.

В процессе обмена ключами могут возникать различные ошибки, в таких ситуациях ответом сервера является сообщение типа KRB_ERROR, которое не шифруется. Отсутствие шифрования сообщений об ошибках делает невозможным отслеживание их повторного использования или их подделку. Сообщения об ошибках содержат

информацию, позволяющую связывать их с соответствующими им запросами.

Обмен с сервером выдачи билетов. Обмен билетами между клиентом и сервером TGS инициируется клиентом, когда он желает получить аутентификационный допуск:

— для некоторого сервера (в общем случае доступ к серверу TGS может осуществляться из удаленного округа);

— при необходимости сделать действительным существующий билет;

— при необходимости получить билет с доверенностью (proxy).

В первом случае клиент должен уже иметь билет TGT, полученный путем AS-обмена, для предъявления серверу TGS. Билет TGT клиент, как правило, получает, когда он аутентифицируется первый раз (при входе в систему).

В табл. 2 представлены пути передачи сообщений TGS-обмена и их типы. Формат сообщений TGS-обмена практически полностью идентичен формату сообщений AS-обмена. Основное отличие заключается в том, что для шифрования и расшифровывания TGS-сообщений не применяется секретный ключ клиента. Вместо него применяется сессионный ключ билета TGT или обновляемого билета, либо субсессионный ключ аутентификатора. При окончании срока действия билета TGT клиент должен инициировать новый обмен для получения действительного билета TGT.

Таблица 2

Сообщения TGS-обмена

Путь прохождения сообщения	Тип сообщения
От клиента к протоколу Kerberos	KRB_TGS_REQ
От протокола Kerberos к клиенту	KRB_TGS_REP KRB_ERROR

Сообщение KRB_TGS_REQ включает ранее полученный билет TGT и аутентификатор, созданный клиентом специально для данного запроса. Аутентификатор, в частности, содержит адрес клиента, метку времени и номер последовательности, используемый для сопоставления запросов и ответов. При создании аутентификатора номер последовательности должен выбираться как случайная величина.

Получив сообщение типа KRB_TGS_REQ, сервер TGS расшифровывает своим секретным ключом билет TGT и извлекает из него сеансовый ключ. После этого с помощью сеансового ключа сервер TGS

расшифровывает аутентификатор и выполняет проверку адреса клиента и метки времени. Данный протокол подразумевает жесткую (до нескольких минут) синхронизацию времени всех участников обмена. Допустимое расхождение значения метки времени и текущего времени сервера определяется политикой округа. Если расхождение превышает некоторое допустимое значение, то такой запрос считается недействительным. Также сервер должен хранить все действительные аутентификаторы, поскольку ложный запрос может появиться в пределах допустимого временно́го интервала.

Использование аутентификаторов объясняется двумя основными причинами: во-первых, аутентификатор содержит некоторую первоначально открытую информацию, зашифрованную сеансовым ключом и позволяющую серверу убедиться, что клиенту известен сеансовый ключ, а следовательно, и его секретный ключ; во-вторых, важно наличие метки времени, которую невозможно незаметно удалить или модифицировать.

При условии успешного прохождения всех проверок сервер TGS создает корректный билет для предоставления серверу приложения и генерирует новый сеансовый ключ, необходимый для CS-обмена. В случае возникновения различных ошибок при обработке запроса клиента ответом сервера является сообщение KRB_ERROR, включающее код ошибки и текст, поясняющий причины произошедшего сбоя, а также информацию, позволяющую устанавливать соответствие между ним и запросом, вызвавшим ошибку. Сообщения данного типа не шифруются.

Аутентификационный обмен типа клиент–сервер. Такой обмен используется сетевыми приложениями для аутентификации клиента и/или сервера. Для участия в данном обмене клиент должен иметь допуск к данному серверу, полученный путем AS- и, возможно, TGS-обмена.

В табл. 3 представлены пути прохождения и типы сообщений CS-обмена. Сообщение типа KRB_AP_REQ содержит, в частности, билет, аутентификатор и дополнительную информацию.

Таблица 3

Сообщения CS-обмена

Путь прохождения сообщения	Тип сообщения
От клиента к серверу приложения	KRB_AP_REQ
От сервера приложения к клиенту (опционально)	KRB_AP_REP KRB_ERROR

Обработка данного сообщения происходит аналогично обработке при TGS-обмене, поскольку сервер TGS является частным случаем сервера ресурсов. При необходимости взаимной аутентификации сервер возвращает аутентификатор, содержащий метку времени и зашифрованный сеансовым ключом.

Специальные сообщения протокола Kerberos. Сообщения типа KRB_SAFE (защищенные) могут использоваться взаимодействующими пользователями, желающими контролировать целостность передаваемой информации и успешно применившими все три субпротокола. Контроль целостности обеспечивается включением в сообщения криптографической контрольной суммы. В сообщениях данного типа используется последний субсеансовый или сеансовый ключ.

Сообщения типа KRB_PRIV (конфиденциальные) могут использоваться взаимодействующими пользователями, желающими защитить данные от несанкционированного доступа и контролировать их целостность. Желаемый результат достигается путем применения криптографических алгоритмов шифрования, ключи для которых выбираются при применении рассмотренных субпротоколов.

Сообщения типа KRB_CRED применяются при пересылке допусков с одного узла на другой. В сообщение включаются билет и зашифрованные данные, содержащие сеансовый ключ и другую необходимую информацию.

Преимущества протокола Kerberos. Каждый аутентифицированный клиент снабжается определенной информацией, которую проверяет сервер приложения. Таким образом, отпадает необходимость в обращении серверов приложений к серверу аутентификации.

Протокол Kerberos позволяет осуществлять взаимную аутентификацию пользователей.

При разработке протокола Kerberos предполагается, что он будет использоваться в открытой сети.

Делегирование аутентификации значительно упрощает допуск к службам от имени субъектов с различными полномочиями.

Доверительные отношения в протоколе Kerberos являются двусторонними и транзитивными.

Протокол Kerberos обеспечивает взаимодействие операционных систем различных платформ.

Отсутствует зависимость от применяемых в протоколе алгоритмов шифрования и хэширования.

Недостатки протокола Kerberos. Недостатки протокола, в основном, связаны с его неспособностью защитить клиентов от атак следующих типов:

— отказов в обслуживании;

— кражи секретных ключей;
— угадывания паролей;
— внедрения программных средств типа “троянский конь”;
— повторного использования аутентификаторов;
— рассогласования времени на компьютерах;
— атаки на незащищенные протоколы, лежащие в основе протокола Kerberos.

Выводы. Протокол сетевой аутентификации Kerberos, безусловно, является одним из самых надежных и эффективных на сегодняшний день. Его эффективность особенно очевидна при применении его в крупных сетях. Однако для данного протокола характерны некоторые недостатки, могущие привести к серьезным последствиям. Необходимо отметить, что система Kerberos постоянно совершенствуется и дорабатывается. В ближайшее время будет произведена интеграция данной системы с инфраструктурой PKI и модернизация управления ключами с помощью криптографических методов с открытым ключом.

СПИСОК ЛИТЕРАТУРЫ

1. Столяров М. А., Трифаленков И. И. На пути к управляемым информационным системам // Jet Info. – 2003. – № 3.
2. Галатенко А. В. О применении методов теории вероятностей для решения задач информационной безопасности. – М.: НИИСИ РАН, 2001.
3. Porras P., Valdes A. Live Traffic Analysis of TCP/IP Gateways // Proc. of the 2002 ISOC Symposium on Network and Distributed Systems Security (2003, July 17).
4. Магауенов Р. Г. Основные задачи и способы обеспечения безопасности автоматизированных систем обработки информации. – М.: Мир безопасности, 1999.
5. Ярочкин В. И. Безопасность информационных систем. – М.: Инфра-М, 2002.

Статья поступила в редакцию 2.06.2004

Николай Викторович Медведев родился в 1954 г., окончил в 1977 г. МВТУ им. Н.Э. Баумана. Канд. техн. наук, зав. кафедрой “Информационная безопасность” МГТУ им. Н.Э. Баумана. Автор 45 научных работ в области исследования и разработки защищенных систем автоматической обработки информации.

N.V. Medvedev (b. 1954) graduated from the Bauman Moscow Higher Technical School in 1977. Ph. D. (Eng.), head of “Data Safety” department of the Bauman Moscow State Technical University. Author of 45 publications in the field of study and development of secured systems of automatic data processing.