Министерство образования Республики Беларусь
Учреждение образования
«Белорусский государственный университет
информатики и радиоэлектроники»

Кафедра сетей и устройств телекоммуникаций

М.Н. Бобов

ПРОТОКОЛЫ АУТЕНТИФИКАЦИИ В СЕТЯХ ТЕЛЕКОММУНИКАЦИЙ

УЧЕБНОЕ ПОСОБИЕ

по курсам «Защита информации в банковских технологиях»,
«Защита программного обеспечения и баз данных в сетях телекоммуникаций",
«Криптографическая защита информации в телекоммуникациях»
для студентов специальности 45 01 03 «Сети телекоммуникаций»
дневной и заочной форм обучения

Минск 2004

УДК 621.395.3(075.8)
ББК 32.882 я 73
Б 72

Р е ц е н з е н т:
доцент кафедры радиотехнических систем БГУИР А.И. Митюхин

Бобов М.Н.

Б 72 Протоколы аутентификации в сетях телекоммуникаций: Учеб. пособие по курсам «Защита информации в банковских технологиях», «Защита программного обеспечения и баз данных в сетях телекоммуникаций", «Криптографическая защита информации в телекоммуникациях» для студ. спец. 45 01 03 «Сети телекоммуникаций» дневной и заочной форм обуч./ М.Н. Бобов.– Мн.: БГУИР, 2004. – 27 с.: ил.
ISBN 985-444-687-5

В учебном пособии рассмотрены наиболее используемые протоколы аутентификации в сетях телекоммуникаций. Изложены принципы функционирования протоколов аутентификации, дана их сравнительная характеристика, проанализированы режимы работы и изложены рекомендации по использованию.

УДК 621.395.3(075.8)
ББК 32.882 я 73

ISBN 985-444-687-5

ВВЕДЕНИЕ

В сетях телекоммуникаций процесс аутентификации включает в себя выполнение двух процедур. Во-первых, при установлении соединения определяется аутентичность объектов, участвующих в сеансе связи, т.е. устанавливается их подлинность, во-вторых, в ходе последующего обмена данными осуществляется защита от влияния на поток данных третьей стороны, пытающейся имитировать одного из законных объектов связи с целью несанкционированной отправки и получения информации.

Реализация указанных процедур аутентификации в телекоммуникационных сетях регламентируется протоколами аутентификации, наиболее используемыми из которых являются:

- парольный протокол аутентификации (Password Authentication Protocol) – PAP;

- протокол аутентификации с согласованным рукопожатием (Challenge Handshake Authentication Protocol) – CHAP;

- расширенный протокол аутентификации (Extensible Authentication Protocol) – EAP;

- протокол аутентификации удаленного пользователя (Remote Authentication Dial-In User Service) – RADIUS;

- протоколы аутентификации в локальных сетях с операционной средой Windows – NT LM и Kerberos.

1. АУТЕНТИФИКАЦИЯ ПОЛЬЗОВАТЕЛЕЙ ЧЕРЕЗ УДАЛЕННОЕ СОЕДИНЕНИЕ

Для аутентификации пользователей через удаленное соединение используются протоколы PAP, CHAP, EAP и RADIUS. Первые три протокола работают совместно с протоколом PPP (Point-to-Point Protocol), а протокол RADIUS в качестве транспорта использует дейтаграммный протокол UDP.

1.1. Описание протокола PAP

Данный протокол - наиболее простой из протоколов подтверждения удаленным субъектом своего идентификатора для объекта, предоставляющего ресурсы для использования. Аутентификация происходит за две итерации. При использовании PAP в поле ***Тип протокола*** кадра PPP указывается соответствующее протоколу PAP значение - 0xC023, поле ***Данные*** преобразуется в четыре дополнительных поля (рис. 1).

Код	Идентификатор	Длина	Данные

Рис. 1. Структура поля ***Данные*** кадра PPP

При этом поле ***Код*** указывает на следующие возможные типы PAP-пакета:

Код = 1 – *Аутентификационный запрос;*

Код = 2 – *Подтверждение аутентификации;*

Код = 3 – *Отказ в аутентификации*.

Поле ***Идентификатор*** обеспечивает соответствие пары запрос/ответ (должен меняться при каждом новом аутентификационном запросе).

Поле ***Длина*** указывает совокупную длину всех четырех полей.

Поле ***Данные*** содержит информацию для аутентификации и в случае запроса имеет вид, приведенный на рис. 2.

Длина идентификатора	Идентификатор	Длина пароля	Пароль

Рис. 2. Структура поля ***Данные*** PAP пакета-запроса

Пакет аутентификационного запроса посылается субъектом, желающим получить доступ, неоднократно до наступления одного из следующих событий:

- получение подтверждения или отказа;
- истечение счетчика попыток.

При получении запроса на аутентификацию объектом производится распознавание полученных результатов (сравнение с имеющимися у объекта значениями). По результатам распознавания субъекту высылается пакет с полем ***Данные*** следующего формата:

Длина сообщения	Сообщение

Рис. 3. Структура поля ***Данные*** PAP пакета-ответа

При этом в полях (см. рис. 1) ***Код*** указывается 2 или 3 (в зависимости от того, подтверждена аутентификация или отвергнута), в поле ***Идентификатор*** – идентификатор соответствующего запроса. В полях ответа (см. рис. 3) указывается: *Длина сообщения* – размер следующего поля, *Сообщение* – возлагается на

конкретную реализацию, оно обязано не влиять на работу протокола и рекомендовано формировать его удобным для прочтения.

Поскольку пакет с подтверждением аутентификации может быть утерян, при реализации протокола необходимо предусматривать возможность обработки повторного запроса на аутентификацию.

Схема работы протокола приведена на рис.4.

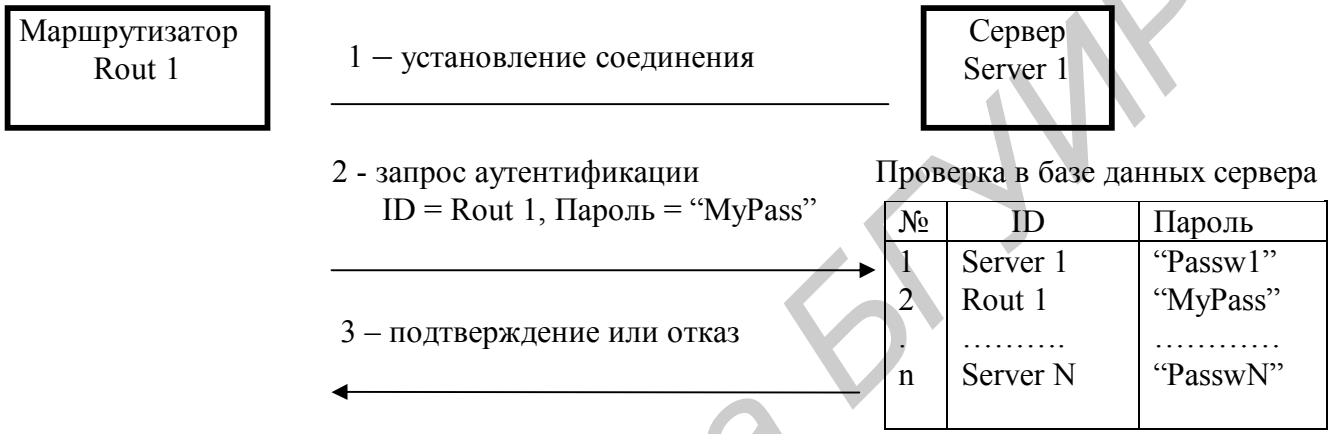

Рис.4. Пример PAP-аутентификации маршрутизатора сервером

Протокол функционирует следующим образом:

1) устанавливается PPP – соединение;

2) субъект посылает аутентификационный запрос с указанием своего идентификатора и пароля;

3) объект проверяет полученные данные и подтверждает аутентификацию или отказывает в ней.

Следует обратить особое внимание, что весь обмен данными (в том числе и пересылка пароля) происходит в открытом виде, без применения криптографических средств. При этом частоту и время отправки пакетов контролирует сам субъект.

1.2. Описание протокола CHAP

Протокол CHAP используется для первоначальной аутентификации субъекта после установления соединения, но может в зависимости от конкретной реализации быть использован для периодического подтверждения подлинности субъекта во время работы в рамках установленного соединения. Аутентификация происходит за три итерации. В поле ***Протокол*** PPP кадра указывается значение 0xC223, поле ***Данные*** преобразуется в четыре поля, аналогично PAP пакету (см. рис. 1) со схожими типами, единственное отличие в том, что поле ***Код*** имеет теперь четыре значения:

Код = 1 – *Запрос на предоставление данных для аутентификации;*

Код = 2 – *Ответ на запрос с аутентификационными данными;*

Код = 3 – *Подтверждение аутентификации;*

Код = 4 – *Отказ в аутентификации*.

Формат поля ***Данные*** зависит от поля ***Код.***

Реализация протокола CHAP требует, чтобы обе стороны имели в распоряжении *заранее согласованный пароль*, который не высылается по сети, но чаще всего хранится у объекта в открытом виде.

Процедура аутентификации в отличие от протокола PAP инициируется не субъектом, а объектом и происходит следующим образом:

1) устанавливается PPP-соединение;

2) объект посылает субъекту запрос на предоставление данных для произведения аутентификации;

3) субъект отвечает необходимыми данными, часть из которых взята из содержимого запроса;

4) объект анализирует полученные данные и отвечает подтверждением или отказом.

Схема работы протокола CHAP приведена на рис.5.

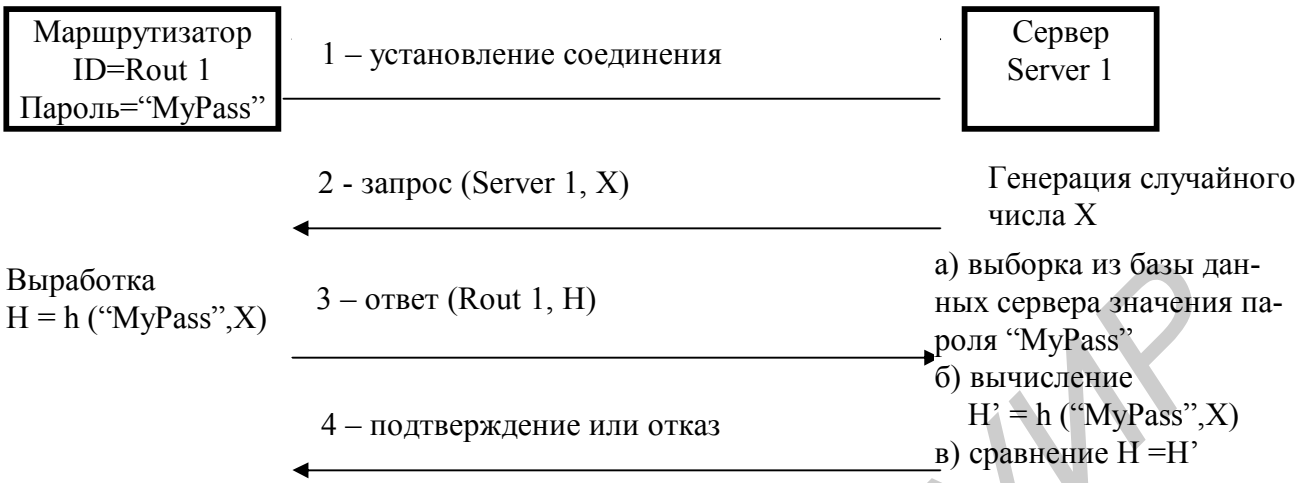

Рис.5. Пример CHAP-аутентификации маршрутизатора сервером

Структура пакета для запроса и ответа (поле ***Данные*** кадра PPP) приведена на рис. 6.

Код	Идентификатор	Длина	Длина числа	Число	Имя

Рис. 6. Структура CHAP-пакета запроса или ответа

В поле ***Код*** указываются значения 1 для запроса и 2 для ответа.

Поле ***Идентификатор*** используется аналогично полю ***Идентификатор*** протокола PAP для обеспечения однозначного соответствия между запросом и ответом. Оно должно уменьшаться при новых запросах.

Поле ***Длина*** показывает суммарную длину всех полей пакета.

Поле ***Длина числа*** определяет размер следующего поля.

В поле ***Число*** указывается специальное значение, которое должно удовлетворять двум требованиям: быть уникальным и случайным. От качества выбора этого числа зависит, сможет ли злоумышленник использовать перехвачен-

ные им запросы-ответы при аутентификации уполномоченного субъекта. Для пакета *Запрос* это значение должно изменяться при каждом новом пакете.

В поле ***Имя*** указывается наименование системы, пославшей пакет. Так как субъект может аутентифицироваться на нескольких объектах, и объект может аутентифицировать несколько субъектов, это поле можно использовать для поиска пароля в соответствующей базе данных.

После получения *Запроса* субъект производит следующие действия:

1) выбирает соответствующий объекту пароль;

2) использует пароль и полученное *Число* как параметры для хэш-функции (обычно MD5) и получает соответствующее хэш-значение;

3) формирует пакет *Ответа,* где в поле ***Число*** указывает полученное хэш-значение, а в поле ***Имя*** – свое наименование;

4) высылает пакет объекту.

Объект, получив пакет, производит свое вычисление хэш-значения с использованием соответствующего пароля для данного субъекта и *Числа.* В случае совпадения рассчитанного и присланного хэш-значения происходит подтверждение аутентификации, иначе – отказ. Формат пакета для этого аналогичен формату пакета PAP с *Кодом* = 3 для успешной авторизации и с *Кодом* = 4 для отказа.

Существует реализация данного протокола компанией Microsoft (MS-CHAP – Microsoft Challenge Handshake Authentication Protocol), в которой хранение пароля субъектом и объектом осуществляется в зашифрованном виде.

1.3. Описание протокола EAP

Главная особенность протокола EAP состоит в том, что в нем могут использоваться различные механизмы опознания, выбор которых переносится на

объект, который может запросить у субъекта дополнительные параметры для определения такого механизма.

В поле ***Протокол*** PPP кадра указывается значение 0xC227, поле ***Данные*** преобразуется в четыре поля, аналогично PAP и CHAP пакету (см. рис. 1) со схожими типами, единственное отличие в том, что поле ***Код*** имеет четыре значения, как и для CHAP:

Код = 1 – *запрос на предоставление данных;*

Код = 2 – *ответ на запрос;*

Код = 3 – *подтверждение аутентификации;*

Код = 4 – *отказ в аутентификации*.

Поскольку процедура аутентификации может потребовать участия пользователя, необходимо предусматривать соответствующие тайм-ауты при рассылках объектом запросов на аутентификацию.

Процедура аутентификации инициируется объектом и происходит следующим образом:

1) устанавливается PPP – соединение;

2) объект посылает субъекту запрос (рис.7) на предоставление данных для произведения аутентификации;

3) субъект отвечает необходимыми данными, часть из которых взята из содержимого запроса;

4) объект анализирует полученные данные и отвечает подтверждением или отказом.

Код	Идентификатор	Длина	Тип	Данные типа

Рис. 7. Структура EAP-пакета запроса или ответа

В поле ***Код*** указываются значения 1 для запроса и 2 для ответа.

Поле ***Идентификатор*** используется аналогично соответствующему полю протокола PAP и CHAP для обеспечения соответствия между запросом и ответом. Оно должно уменьшаться при новых запросах. Однако необходимо учесть, что поле ***Идентификатор*** может содержать повторяющееся значение (и должно совпадать для повторяющихся запросов) для рассмотренных случаев запаздывания с получением данных от пользователя.

Поле ***Длина*** указывает суммарную длину всех полей пакета.

Поле ***Тип*** обозначает тип запроса или ответа (собственно тип аутентификации) и должно указываться в запросе и совпадать с ним в ответе (если субъект поддерживает предложенный тип аутентификации), либо в ответе указывается NAK, если данный тип аутентификации не поддерживается. (Субъект также может указать приемлемый для него тип аутентификации).

Поле ***Данные типа*** содержит данные, соответствующие указанному типу.

При успешной аутентификации объект высылает субъекту пакет следующей структуры (рис. 8).

Код	Идентификатор	Длина

Рис. 8. Структура EAP-пакета подтверждения или отказа аутентификации

Поле ***Код*** равно 3 при подтверждении и 4 - при отказе. Однако при этом объект может запросить дополнительную аутентификацию, учитывая, что отказ мог произойти по причине человеческой ошибки.

Поле ***Идентификатор*** соответствует идентификатору запроса, на который посылается ответ.

Поле ***Длина*** равно 4.

Теперь рассмотрим, какие типы аутентификации (значения поля ***Тип*** в запросе/ответе) поддерживаются протоколом:

Тип = 1 – *обозначение субъекта*, используется для определения субъекта, возможно, в поле ***Данные типа*** использование сообщения для приглашения пользователя ко вводу своего идентификатора (не путать с идентификатором пакета);

Тип = 2 – *сообщение,* используется для передачи субъекту сообщения (например, сообщение о скором истечении пароля);

Тип = 3 – *NAK* , используется только при ответе и указывает, что предложенный метод аутентификации неприемлем для субъекта, в поле ***Данные типа*** указывается желаемый тип аутентификации;

Тип = 4 – *ответ MD5,* используется для запроса/ответа (число или хэш-значение) согласно протоколу CHAP;

Тип = 5 – *одноразовый пароль,* для данного типа поле ***Данные типа*** содержит запрос для одноразового пароля или 6 слов из соответствующего словаря одноразовых паролей;

Тип = 6 – *вид электронного ключа*, используется при применении различных типов электронных ключей, требующих пользовательского представления. В запросе указывается текстовая (ASCII) информация, а в ответе – аутентификационные данные (например, то, что пользователь считал с электронного ключа).

1.4. Описание протокола RADIUS

RADIUS - это клиент/серверный протокол, где в качестве клиента обычно выступает *сервер сетевого доступа* (NAS – Network Access Server), с которым соединяется субъект-пользователь, а в качестве сервера – объект (авторизованный сервер). RADIUS аутентифицирует транзакции на основе общего секрета между ним и NAS, не передаваемого по сети, и шифрует им же пароли пользователей при пересылке между клиентом и сервером.

Один RADIUS-сервер может выступать в качестве посредника между клиентом и другим RADIUS-сервером, т.е. направлять второму серверу запросы клиента и возвращать клиенту ответы второго сервера.

Когда пользователь пытается получить доступ к объекту через клиента (в нашем случае NAS), который использует для аутентификации RADIUS-сервер, то он должен предоставить данному клиенту необходимую аутентификационную информацию. Клиент формирует *Запрос* на доступ (рис.9), содержащий имя пользователя, его пароль, идентификатор клиента и идентификатор порта, к которому пользователь пытается получить доступ. При этом пароль шифруется по алгоритму MD5 по схеме со сцеплением блоков (CFB). В качестве начального вектора используется уникальный номер *Запроса,* а в качестве ключа – заранее установленный между NAS и RADIUS *Секретный ключ.*

Данный *Запрос* направляется RADIUS-серверу, причем повторяется несколько раз, если ответ от сервера не приходит в заданное время. *Запросы* могут отправляться другим серверам, если первичный сервер дал сбой.

При получении *Запроса* сервером он идентифицирует клиента и не обрабатывает *Запросы* от клиентов, с которыми у него нет общего *Секретного ключа.* Если клиент одобрен, происходит анализ имени пользователя, указанного в *Запросе.* Сервер просматривает базу данных пользователей на предмет информации о данном пользователе, такой как пароль, идентификатор клиента и порта, с которыми ему разрешено работать. При этом сервер может обратиться к другим серверам за необходимой информацией (в этом случае он сам выступает в роли RADIUS-клиента). Если же в *Запросе* присутствуют атрибуты, указывающие, что RADIUS-сервер должен выступить в качестве посредника, то вся информация из *Запроса* без изменения копируется в соответствующий пакет.

Если условия, необходимые для анализа, не выполняются, сервер формирует и отсылает ответ *Отказ в доступе*, при необходимости сопровождая его

соответствующей текстовой информацией, которая будет отображена пользователю.

При соблюдении условий сервер может сразу аутентифицировать клиента пакетом *Подтверждение доступа* либо потребовать от него дополнительную информацию, формируя в этом случае *Требование по доступу*, которое может содержать текстовую информацию для пользователя, необходимую для ответа на данный вопрос. Необходимо учесть, что RADIUS поддерживает аутентификацию пользователей по протоколам PAP и CHAP. Клиент, получив *Требование*, ожидает от пользователя ответ, затем формирует новый *Запрос* (с новым идентификатором запроса), а атрибут пароля заменяет шифрованным *Ответом пользователя* и включает атрибут *Статуса* (если таковой присутствовал в *Требовании*).

Сервер отвечает на данный запрос либо пакетом *Подтверждение доступа* либо пакетом *Отказ в доступе,* либо новым пакетом *Требование к доступу* (рис. 10). Если производится подтверждение доступа, то в пакет вносятся дополнительные атрибуты, необходимые для предоставления сервиса (например, IP-адрес, маска подсети и прочие возможные параметры).

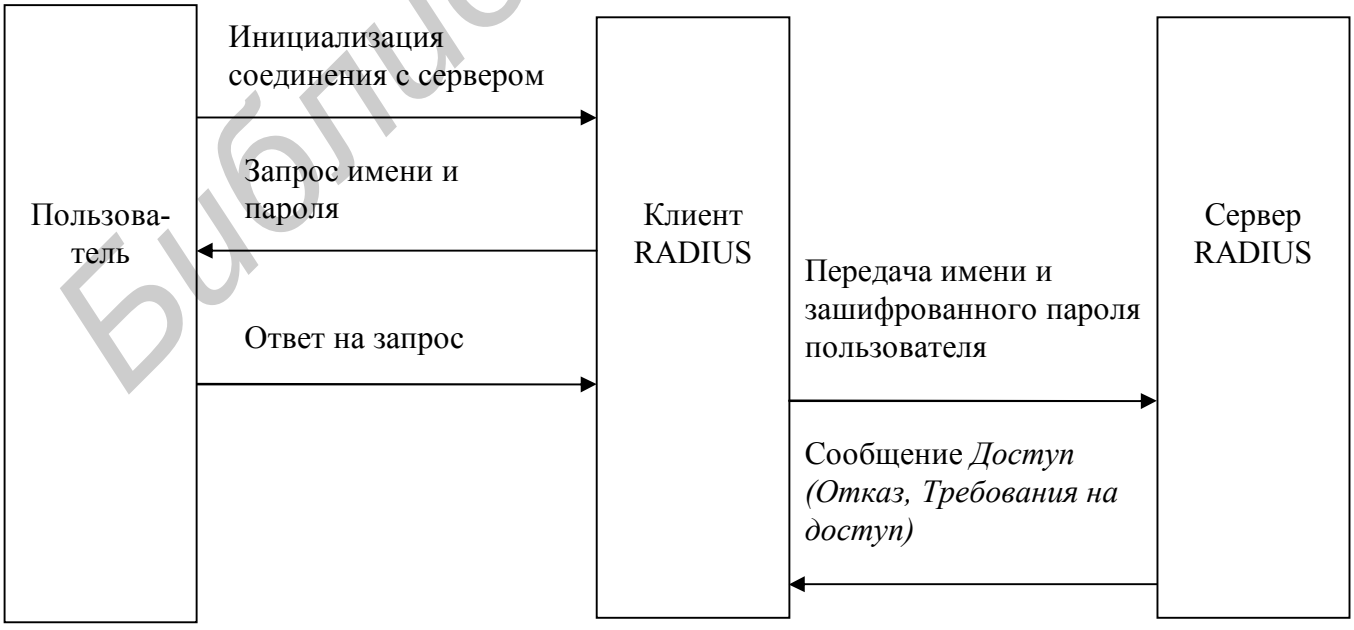

Рис.9. Схема взаимодействия пользователя с системой RADIUS

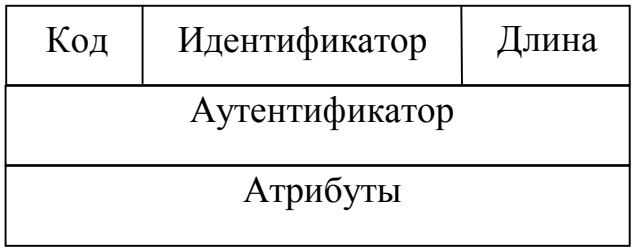

Рис. 10. Общая структура пакета RADIUS

Поле ***Код*** предназначено для указания типа пакета. В настоящее время определены следующие типы пакетов: *Запрос на доступ, Подтверждение доступа, Отказ в доступе, Запрос по учету, Ответ по учету, Требование по доступу, Статус сервера, Статус клиента*.

Поле ***Идентификатор*** предназначено для обеспечения соответствия запрос/ответ.

Поле ***Длина*** указывает длину всего пакета, включая поля ***Код, Идентификатор, Длина, Аутентификатор*** и ***Атрибуты***, и может указывать длину от 20 до 4096 байт. Если реально пакет меньше указанной длины, он не будет обработан. Все данные, превышающие указанную длину, не будут обработаны.

Поле ***Аутентификатор*** имеет различные назначения. Для запросов это – *Аутентификатор запроса* – случайное число, используемое для обеспечения защиты от повторного использования перехваченных пакетов. Это значение в дальнейшем используется в качестве начального вектора схемы шифрования CFB и указывается в атрибутах. Для пакетов ***Подтверждение доступа*** или ***Отказ в доступе*** и ***Требование по доступу*** это – *Аутентификатор ответа*, представляющий собой следующее значение MD5 (Код + Идентификатор + Длина + Аутентификатор запроса + Атрибуты + Общий секретный ключ), при

этом знак "+" означает конкатенацию ("склеивание") соответствующих блоков данных.

Поле ***Атрибуты*** переменной длины представляет собой набор произвольного количества пар "Тип/Значение", которые содержат дополнительные данные, требующиеся для различных сервисов и прочих целей. Его формат приведен на рис. 11.

Рис. 11. Формат поля ***Атрибуты***

Поле ***Тип*** используется для указания типа атрибута, которыми могут являться: *Имя пользователя, Пароль пользователя* (в открытом виде), *Пароль пользователя (схема аутентификации CHAP), IP адрес NAS, Тип сервиса, Текстовое сообщение* – краткий ответ пользователю, *Двоичные параметры сессии* в нерегламентированном формате, *Максимальная продолжительность сессии* в секундах, *Максимальное возможное бездействие в пределах сессии* в секундах, *Телефонный номер*, набранный пользователем, *Телефонный номер пользователя* (в случае, если он определен), *Статус посредника* – служебная информация, добавляемая RADIUS-сервером в тех случаях, когда он играет роль клиента другого RADIUS-сервера; *Двоичные данные CHAP-запроса* в тех случаях, когда сервер по каким-либо причинам не может разместить их в поле ***Аутентификатор.***

Поле ***Длина*** указывает длину записи об атрибуте, включая *Тип, Длину* и *Значение*.

Поле ***Значение*** (переменной длины) содержит значение атрибута и может быть одним из пяти типов: *text* (текст), *string* (двоичные данные переменной

длины), *address* (32-битное целое – адрес), *integer* (32-битное целое) и *time* (32-битное целое – время в формате UNIX).

Таким образом, транзакции между клиентом и сервером RADIUS аутентифицируются с помощью общего секретного ключа, который никогда не передается по сети. Кроме того, обмен любыми пользовательскими паролями между клиентом и сервером RADIUS происходит только в зашифрованном виде, что исключает подслушивание чужих паролей и последующее их несанкционированное использование.

2. ПРОТОКОЛЫ АУТЕНТИФИКАЦИИ В ЛОКАЛЬНЫХ СЕТЯХ С ОПЕРАЦИОННОЙ СРЕДОЙ WINDOWS

Процесс аутентификации в локальной сети осуществляется в следующих случаях:

а) после включения компьютера и загрузки операционной системы;

б) при обращении к услугам необходимых сервисов;

в) после блокировки интерактивного сеанса пользователя самим пользователем.

Идентификация и аутентификация пользователя с рабочей станции, на которой установлена ОС Windows NT, выполняются с использованием протокола NTLM, а с рабочей станции, на которой установлена ОС Windows 2000, - с использованием протокола Kerberos.

В сетях с операционной средой Windows существует два типа аутентификации: интерактивная и неинтерактивная. Интерактивная сетевая аутентификация предполагает использование двух систем: системы клиента, с которой пользователь регистрируется, и контроллера домена, на котором хранится информация, относящаяся к паролю пользователя. Неинтерактивная аутентификация, которая может потребоваться для обеспечения уже зарегистрированному пользователю

доступа к защищенным ресурсам, например на сервере приложений, обычно использует три системы: клиента, сервера и контроллера домена, который выполняет необходимые для аутентификации операции от имени сервера.

Компоненты, используемые в процессах аутентификации в сетях с операционной средой Windows, и их взаимодействие изображены на схеме, приведенной на рис. 12.

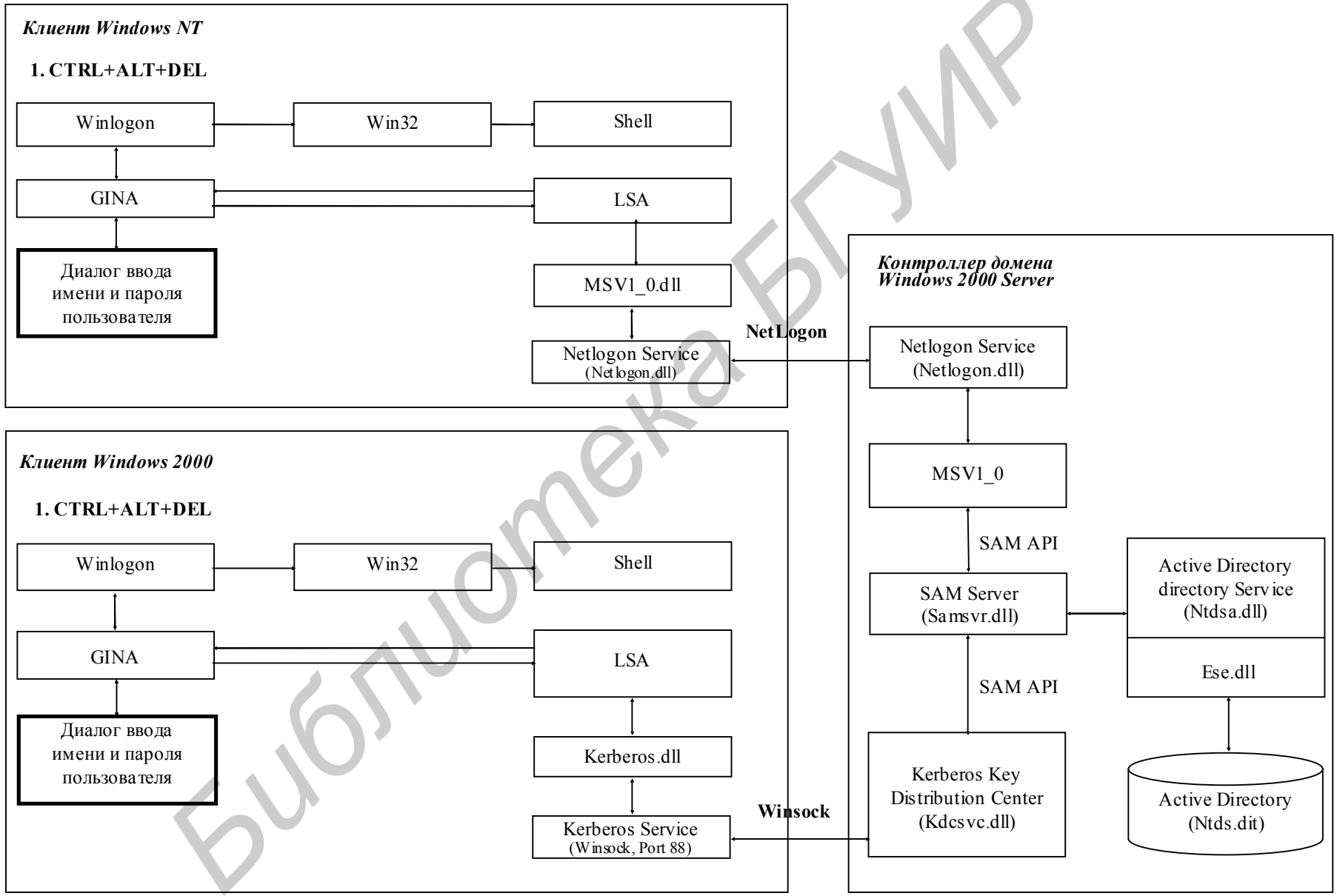

Рис. 12. Схема взаимодействия средств идентификации и аутентификации

2.1. Аутентификация с использованием протокола NTLM

Интерактивная сетевая аутентификация с помощью NTLM осуществляется следующим образом. После того, как пользователь, используя графический интерфейс идентификации и аутентификации (GINA), вводит свое имя и пароль, система аутентифицирует его путем передачи параметров, заданных в вводном диалоговом окне, менеджеру защиты учетных записей SAM (Security Account Manager). Передача параметров идентификации и аутентификации осуществляется с помощью диспетчера локальной безопасности (LSA), который обеспечивает согласование системных процедур аутентификации с конкретными реализациями протоколов взаимодействия. SAM сравнивает имя пользователя и зашифрованный пароль с теми, что хранятся в базе пользователей домена или, при локальной регистрации, рабочей станции. Если имя и пароль совпадают, сервер уведомляет рабочую станцию о подтверждении доступа. Если пользователь имеет учетную запись и привилегии доступа в систему, а введенный им пароль верен, подсистема защиты создает объект *маркер доступа,* представляющий пользователя. Он сравним с ключом, содержащим «удостоверение личности» пользователя. *Маркер доступа* хранит такую информацию, как идентификатор защиты SID (Security ID), имя пользователя и имена групп, к которым он принадлежит.

Созданный маркер передается процессу Win32 WinLogon, который предписывает подсистеме Win32 создать процесс для пользователя, и *маркер доступа* присоединяется к этому процессу. После этого подсистема Win32 инициирует Windows NT Explorer и на экране появляется соответствующее окно.

Процесс неинтерактивной аутентификации осуществляется на основе протокола NTLM, в котором используется метод «запрос-ответ». Идентификационные и аутентификационные параметры (мандат) NTLM основаны на данных, получаемых в процессе интерактивного процесса регистрации, и состоят из имени

домена, имени пользователя и хэш-значения пароля пользователя. NTLM использует шифрованный протокол «вызов/ответ» для аутентификации пользователя без пересылки пароля пользователя по сети. При этом система, вызвавшая процесс аутентификации, выполняет соответствующие вычисления, которые подтверждают подлинность субъекта, получающего доступ к защищенным объектам.

Протокол NTLM реализован в модуле MSV1_0.dll и функционирует следующим образом:

1) пользователь получает доступ к клиентской машине и указывает имя домена, имя пользователя и пароль. Клиентская система вычисляет хэш-значение пароля и уничтожает пароль, введенный пользователем (получение мандата NTLM пользователя выполняется как часть процесса интерактивной аутентификации);

2) клиент посылает имя пользователя на сервер (в текстовом виде);

3) сервер генерирует 16-байтное случайное число, называемое «Запрос», и отсылает его клиенту;

4) клиент шифрует этот запрос хэш-значением пароля пользователя и возвращает результат на сервер. Этот шаг называется «Ответ»;

5) сервер отсылает контроллеру домена следующие три значения: имя пользователя, запрос, посланный клиенту, и ответ, полученный от клиента;

6) контроллер домена использует имя пользователя для получения хэш-значения пароля пользователя из базы данных SAM, затем он использует полученное хэш-значение для зашифрования запроса;

7) контроллер домена сравнивает полученный им зашифрованный запрос (п. 6) с ответом, полученным от клиента (п. 4). Если они идентичны, аутентификация считается успешной.

2.2. Аутентификация с использованием протокола Kerberos

Система Kerberos представляет собой службу KDC (Key Distribution Center – Центр распределения ключей), выполняющуюся на контроллере домена под управлением Windows 2000 Server и использующую Active Directory как базу данных учетных записей. KDC – единый процесс, обеспечивающий работу следующих сервисов:

а) сервис аутентификации (Authentication Service – AS) – предназначен для формирования TGT (Ticket Granting Ticket – билет, обеспечивающий получение билетов), необходимого для подключения к сервису выдачи билетов. Перед тем, как клиент сможет запросить билет для доступа к другим компьютерам домена, он должен запросить TGT от AS домена, к которому принадлежит клиент. AS возвращает TGT для сервиса выдачи билетов в текущем домене. TGT может быть использован многократно до истечения срока его действия, но для доступа к сервису выдачи билетов в первый раз всегда необходимо выполнить запрос к AS домена;

б) сервис выдачи билетов (Ticket-Granting Service – TGS) – предназначен для формирования билетов для доступа к компьютерам в их собственных доменах. Когда клиенту нужно получить доступ к компьютеру, он делает запрос к TGS, работающему в домене, к которому принадлежит компьютер, предоставляет свой TGT и запрашивает билет к нужному компьютеру. Билет может быть использован многократно до истечения срока его действия, но для доступа к компьютеру в первый раз всегда необходимо выполнить запрос к TGS целевого домена.

В процессе функционирования системы Kerberos осуществляются несколько видов обмена сообщениями в зависимости от требуемых функций.

А. Обмен с сервисом начальной аутентификации (рис. 13).

Общение с сервисом начальной аутентификации обычно инициируется клиентом, желающим получить удостоверение к определенному серверу, но не имеющим пока других удостоверений. Для шифрования и расшифрования используется секретный ключ клиента. Как правило, данный обмен происходит при входе в систему для получения удостоверения к сервису выдачи билетов. Кроме того, общение с сервисом начальной аутентификации используется для получения удостоверения к серверам, требующим знания именно секретного ключа (пример – сервер смены пароля).

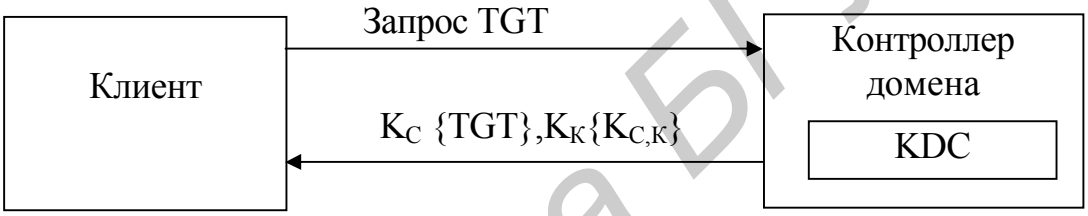

Рис. 13. Схема взаимодействия при начальной аутентификации

В своем запросе (KRB_AS_REQ) клиент посылает открытым текстом свое имя и имя сервиса, к которому он хочет получить удостоверение. Ответ (KRB_AS_REP) содержит билет (Ticket Granting Ticket – TGT), который клиент должен будет предоставить сервису выдачи билетов, и сеансовый ключ $K_{C,K}$ для совместного использования клиентом и KDC. Билет шифруется секретным ключом KDC K_C, сеансовый ключ и дополнительная информация — секретным ключом клиента K_K. Сообщение KRB_AS_ REP содержит данные, позволяющие связать его с предыдущим запросом (KRB_AS_REQ) и обнаружить дублирование сообщений. В случае какой-либо ошибки возвращается сообщение KRB_ERROR, которое не шифруется.

Сервис аутентификации не делает попыток убедиться в подлинности обратившегося к нему субъекта. Он просто возвращает информацию, воспользоваться которой может лишь тот, кто знает секретный ключ субъекта.

Б. Обмен с сервисом выдачи билетов (TGS).

Обмен между клиентом и сервисом выдачи билетов инициируется клиентом, когда тот хочет получить удостоверение к определенному серверу или компьютеру (возможно, зарегистрированному в удаленной области управления), обновить или зарегистрировать существующий билет. Клиент должен располагать предварительно полученным от сервиса начальной аутентификации билетом к TGS. Формат сообщений при обмене с сервисом выдачи билетов почти тот же, что и для сервиса начальной аутентификации. Основное отличие состоит в том, что для шифрования и расшифрования используется сеансовый ключ.

Запрос (KRB_TGS_REQ) состоит из информации, подтверждающей подлинность клиента (TGT), и заявки на удостоверение. Ответ (KRB_TGS_REP) содержит запрашиваемое удостоверение, зашифрованное сеансовым ключом, и данные, позволяющие обнаружить дублирование сообщений и связать ответ с запросом.

В. Аутентификационный обмен «клиент-сервер».

Данный обмен используется сетевыми приложениями для взаимной проверки подлинности. Клиент должен располагать предварительно полученным удостоверением к серверу. Он передает серверу билет, аутентификатор (зашифрованный сеансовым ключом) и некоторую дополнительную учетную информацию. Сервер в ответ возвращает только аутентификатор, зашифрованный тем же сеансовым ключом.

Чтобы с помощью Kerberos получить доступ к серверу S (рис. 14), клиент C посылает системе Kerberos запрос, содержащий сведения о нем (клиенте) и о запрашиваемой услуге. В ответ Kerberos возвращает информацию двух видов:

билет, зашифрованный секретным ключом сервера K_S, и копию части информации из билета, зашифрованную секретным ключом клиента. Клиент должен расшифровать вторую часть данных и переслать ее вместе с билетом серверу. Сервер, расшифровав билет, может сравнить его (билета) содержимое с дополнительной информацией, присланной клиентом. Совпадение свидетельствует о том, что клиент смог расшифровать предназначенные ему данные (ведь содержимое билета никому, кроме сервера и Kerberos, недоступно), т.е. продемонстрировал знание своего секретного ключа. Значит, клиент — именно тот, за кого себя выдает.

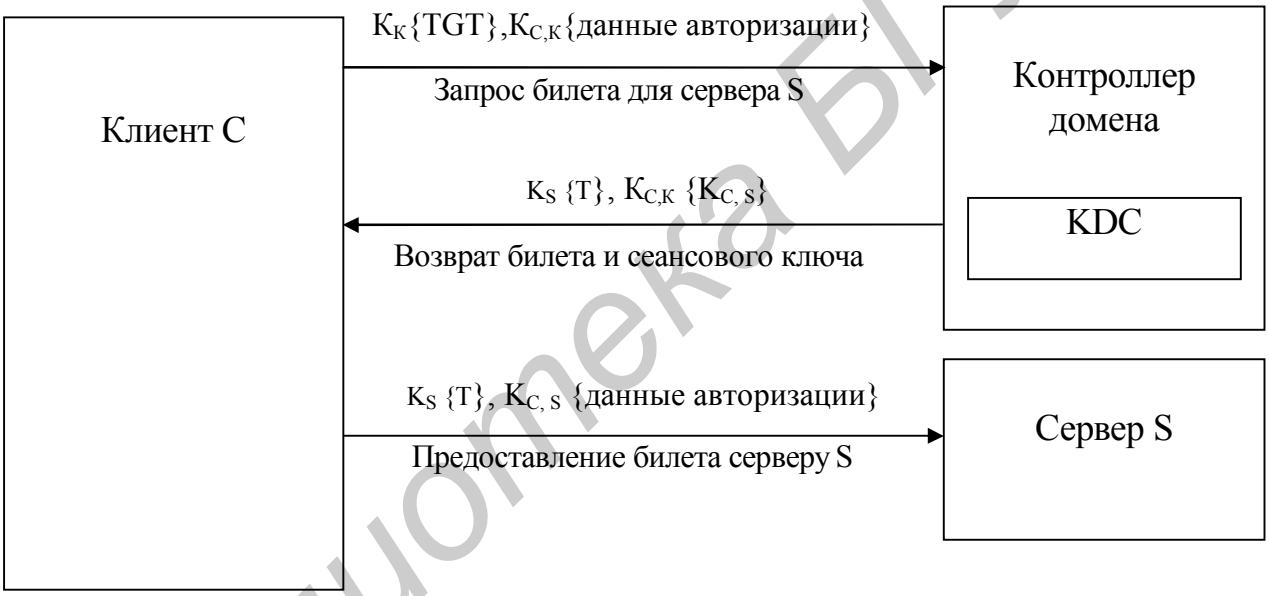

Рис. 14. Схема аутентификационного обмена «клиент – сервер»

Следует подчеркнуть, что секретные ключи в процессе проверки подлинности не передаются по сети (даже в зашифрованном виде) — они только используются для шифрования.

КОНТРОЛЬНЫЕ ВОПРОСЫ И ЗАДАЧИ

1. Как осуществляется интерактивная аутентификация пользователей по протоколу NTLM ?

2. В чем отличие протоколов аутентификации PAP и CHAP ?

3. Разработать алгоритм реализации протокола RADIUS.

4. Пояснить сущность и особенности протоколов аутентификации в соединениях «точка-точка».

5. Как определяется подлинность пользователя при обращении к сервисам в сетях с операционной средой Windows 2000?

ЛИТЕРАТУРА

1. Иванов М.А. Криптографические методы защиты информации в ко терных системах и сетях. - М.: КУДИЦ - ОБРАЗ, 2001.- 368 с.

2. Романец Ю.В., Тимофеев П.А., Шаньгин В.Ф. Защита информации в компьютерных системах и сетях. 2-е изд. – М.: Радио и связь, 2002. – 328 с.

3. Стенг Д., Мун С. Секреты безопасности сетей. - Киев: Диалектика, 1996.- 544 с.

4. Гайкович В., Першин А. Безопасность электронных банковских систем/ Под ред. Ю.В. Гайковича. – М.: Единая Европа, 1994.- 363 с.

5. Олифер В.Г., Олифер Н.А. Компьютерные сети. Принципы, технологии, протоколы: Учебник для вузов. - СПб.: Питер, 2001. – 672 с.

6. Кульгин М. Технологии корпоративных сетей: Энциклопедия. – СПб.: Питер, 2000. – 704 с.

Св. план 2004, поз.109

Учебное издание

Бобов Михаил Никитич

ПРОТОКОЛЫ АУТЕНТИФИКАЦИИ В СЕТЯХ ТЕЛЕКОММУНИКАЦИЙ

УЧЕБНОЕ ПОСОБИЕ

по курсам «Защита информации в банковских технологиях»,
«Защита программного обеспечения и баз данных в сетях телекоммуникаций",
«Криптографическая защита информации в телекоммуникациях»
для студентов специальности 45 01 03 «Сети телекоммуникаций»
дневной и заочной форм обучения

Редактор Н.А. Бебель

Подписано в печать 07.09.2004. Формат 60x84 1/16. Бумага офсетная.
Печать ризографическая. Гарнитура «Таймс». Усл. печ. л. 1,74.
Уч.-изд. л. 1,5. Тираж 50 экз. Заказ 225.

Издатель и полиграфическое исполнение: Учреждение образования
«Белорусский государственный университет информатики и радиоэлектроники»
Лицензия на осуществление издательской деятельности №02330/0056964 от 01.04.2004.
Лицензия на осуществление полиграфической деятельности №02330/0133108 от 30.04.2004.
220013, Минск, П. Бровки, 6