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

Кафедра сетей и устройств телекоммуникаций

М.Н. Бобов

ИМИТОЗАЩИТА СООБЩЕНИЙ В СЕТЯХ СВЯЗИ

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 2005

УДК 621.395.3 (075.8)
ББК 32.882 я 73
Б 72

Р е ц е н з е н т:
доцент кафедры систем телекоммуникаций,
канд.техн.наук О.А. Хацкевич

Бобов М.Н.

Б 72 Имитозащита сообщений в сетях связи: 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, 2005. – 24 с.: ил.
ISBN 985-444-774-X

В учебном пособии рассмотрены наиболее используемые методы имитозащиты сообщений в сетях связи. Представлены алгоритмы реализации методов имитозащиты сообщений, дана их сравнительная характеристика, проанализированы режимы работы и изложены рекомендации по использованию.

УДК 621.395.3 (075.8)
ББК 32.882 я 73

ISBN 985-444-774-X

СОДЕРЖАНИЕ

ВВЕДЕНИЕ

Механизм имитозащиты сообщений в телекоммуникационных сетях (ТКС) реализуется с помощью методов аутентификации передаваемых сообщений. Методы аутентификации сообщений основаны на анализе приемной стороной проверочных комбинаций, которые формируются и добавляются к защищаемым сообщениям в соответствии с определенными правилами на передающей стороне.

Под аутентификацией сообщений понимают установление их подлинности исключительно на основе содержащейся в них информации, включающее установление законным получателем (возможно арбитром) того факта, что данная полученная информация наиболее вероятно была передана законным отправителем (источником) и что при этом она не заменена и не искажена. Любые преднамеренные и случайные нарушения подлинности обнаруживаются в соответствии с заданной вероятностью.

Целью аутентификации информации в ТКС является обеспечение защиты участников информационного обмена от обмана, осуществляемого нарушителем путем навязывания ложной информации как при условии наличия взаимного доверия между участниками данного информационного обмена, так и при отсутствии его. В связи с этим условием различают классическую аутентификацию информации и аутентификацию в широком смысле.

Методы классической аутентификации информации, в предположении наличия взаимного доверия между участниками этого информационного обмена, известны в литературе как методы обеспечения имитозащиты с помощью симметричных криптографических алгоритмов.

При отсутствии взаимного доверия между участниками информационного обмена методы классической аутентификации информации дают, как правило, громоздкие технические решения, а в отдельных случаях и вовсе неприменимы. Поэтому в таких условиях используются несимметричные криптографи-

ческие алгоритмы, а аутентификация информации включает в себя и аутентификацию абонентов, например цифровую подпись.

В модели аутентификации сообщений представлены четыре участника. Это отправитель (законный пользователь А), получатель (законный пользователь В), нарушитель в канале связи (С) и доверенная сторона (Д). Исходное распределение функций между участниками модели следующее.

Задача пользователя А заключается в отправке сообщения X пользователю В. Задача пользователя В заключается в получении сообщения X и в проверке его подлинности. Задачей доверенной стороны является обеспечение функционирования защищенного тракта обмена ключевой системы, например: рассылка ключей, восстановление после компрометации секретных ключей, запись в своей защищенной памяти различной информации о функционировании сети.

Рассмотрим различные способы обмана (нарушение подлинности сообщения), которые возможны в условиях данной модели аутентификации сообщений, с учетом того, что в действиях возможных нарушителей из числа участников модели А, В и С отсутствует кооперация.

Способ C_1^0: нарушитель в канале связи искажает сообщение (возможно, после проведения анализа на основе этого и других сообщений), которое пользователь А передает пользователю В (подмена передаваемого сообщения).

Способ C_2^0: нарушитель в канале связи формирует и посылает пользователю В сообщение X от имени пользователя А (имитация передаваемого сообщения).

Способ C_3^0: нарушитель в канале связи повторяет ранее переданное сообщение, которое пользователь А посылал пользователю В (повтор ранее переданного сообщения).

Имитозащита сообщений подразумевает защиту против нарушителя в канале связи, т.е. против способов обмана C_1^0, C_2^0, C_3^0.

1. ПРИНЦИПЫ РЕАЛИЗАЦИИ МЕТОДОВ ИМИТОЗАЩИТЫ

В литературе описаны два метода классической аутентификации: метод MAC (message anthentication code − код аутентификации сообщений) и метод MDC (manipulation detection code – код обнаружения имитаций).

Проверочная комбинация может быть вычислена как криптографическая функция от сообщения X и секретного ключа K, известного только взаимодействующим пользователям (в нашем случае пользователям A и B) и, возможно, доверенной стороне.

1.1. Метод MAC

В этом методе пользователь A вычисляет проверочную комбинацию (имитовставку) сообщения X:

$$Z = F(k,X),$$

используя один из симметричных алгоритмов шифрования и секретный ключ. Полученная имитовставка присоединяется к сообщению и отправляется пользователю B, т.е. в виде

$$\overline{X} = (X,Z).$$

Заметим, что в состав сообщения X входят как данные, подлежащие передаче, так и служебная информация, например: адрес отправителя, адрес получателя, дата, время отправки, номер сообщения.

Пользователь B получает $\overline{X}$, вычисляет $Z^* = F(k,X)$, затем удостоверяется в том, что результат этого вычисления совпадает с полученным:

$$Z^* = Z.$$

Если предположить, что секретный ключ знают только отправитель и получатель, и если выполнено последнее равенство, то можно утверждать следующее:

1. Получатель может быть уверен, что сообщение не было изменено. Если нарушитель изменит сообщение, но не изменит соответствующим образом значение MAC, то вычисленное получателем значение будет отличаться от пришедшего с сообщением. Не зная секретного ключа, нарушитель не сможет правильно изменить значение MAC, чтобы оно соответствовало измененному сообщению.

2. Получатель может быть уверен, что сообщение пришло от указанного отправителя. Поскольку секретный ключ никому, кроме указанного отправителя, неизвестен, никто другой не может подготовить сообщение с соответствующим значением MAC.

3. Если сообщение включает присвоенный ему порядковый номер, то получатель может быть уверен в том, что сообщения приходят в правильной последовательности, поскольку нарушитель не может изменить порядковый номер сообщения.

Для формирования MAC используются известные алгоритмы симметричного шифрования, где контрольная комбинация вырабатывается в режиме со сцеплением блоков.

В алгоритме ГОСТ 28147-89 для вычисления MAC предусмотрен специальный режим вычисления имитовставки, который включает в себя следующие операции:

1. Первые 64 бита сообщения подвергаются преобразованию, соответствующему первым 16 циклам алгоритма шифрования в режиме простой замены:

$$Z_{64}^{1} = E_K(X_{64}^{1}).$$

2. Полученное значение суммируется по модулю два со вторым блоком сообщения:

$$Z_{64}^{1} + X_{64}^{2} = Y_{64}^{1}.$$

3. Результат суммирования подвергается преобразованию, соответствующему первым 16 циклам алгоритма зашифрования в режиме простой замены:

$$Z_{64}^{2} = E_K(Y_{64}^{1}).$$

4. Полученный результат суммируется по модулю два с третьим блоком сообщения:

$$Z_{64}^{2} \oplus X_{64}^{3} = Y_{64}^{2}.$$

5. Действия по пп. 3, 4 повторяются до выборки последнего блока сообщения.

6. Последний блок сообщения, при необходимости дополненный до полного 64-битного блока нулями, суммируется по модулю два с предыдущим результатом шифрования:

$$Z_{64}^{n-1} \oplus X_{64}^{n} = Y_{64}^{n-1}.$$

7. Результат суммирования зашифровывается в режиме простой замены по первым 16 циклам работы алгоритма:

$$Z_{64}^{n} = E_K(Y_{64}^{n-1}).$$

8. Из полученного результата зашифрования выбирается отрезок U_L (имитовставка) длиной L бит.

Вероятность навязывания ложного сообщения в этом случае равна

$$P_H = 2^{-L}.$$

1.2. Метод MDC

Проверочная комбинация может быть вычислена на основе использования необратимой функции сжатия сообщения X – хеш-функции.

Как и в случае метода MAC, функция хеширования получает на вход сообщение X произвольной длины, а на выходе выдает хеш-код h = H(X) фиксированной длины. В отличие от значения MAC функция хеширования не требует использования секретного ключа. Существует несколько вариантов реализации метода MDC.

На рис.1 представлена наиболее распространенная схема реализации метода MDC.

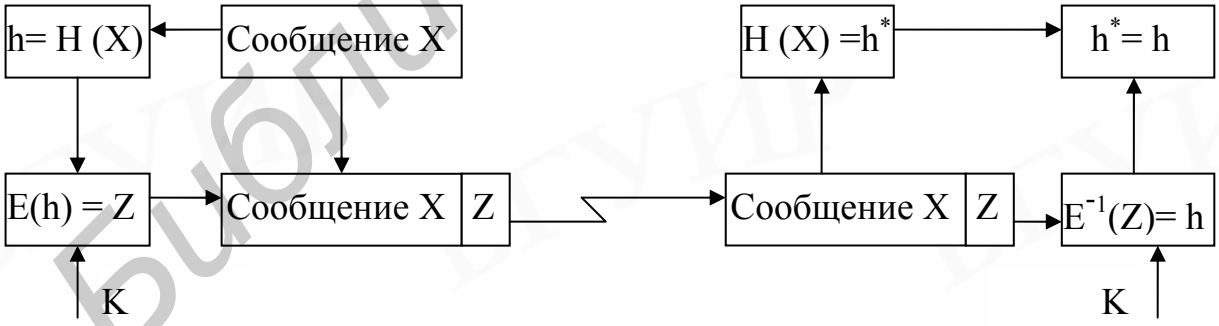

Рис. 1. Схема реализации метода MDC

В этом методе пользователь А вычисляет хеш-код сообщения X, зашифровывает значение хеш-кода с использованием симметричного алгоритма, при-

соединяет зашифрованное значение хеш-кода к сообщению и отправляет полученный блок пользователю В. Пользователь В расшифровывает полученный хеш-код, вычисляет хеш-код сообщения X, сверяет результат этого вычисления с расшифрованным значением полученного хеш-кода и по результату сравнения делает вывод о подлинности полученного сообщения.

Схема второго варианта использования метода MDC приведена на рис.2.

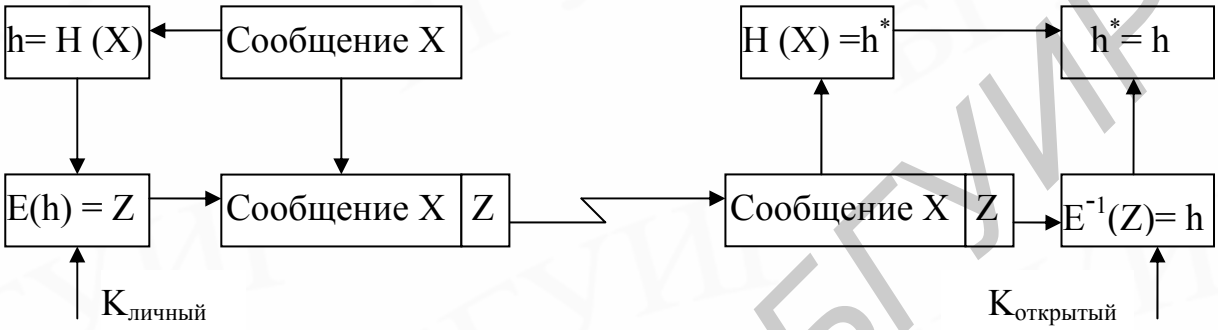

Рис.2. Схема реализации метода MDC с использованием ассиметричного алгоритма шифрования

В данном варианте получателю не требуется знание секретного ключа, что избавляет от необходимости доставки секретного ключа сообщающимся сторонам.

Третий вариант метода MDC предполагает использование функции хеширования без зашифрования хеш-кода. Схема третьего варианта реализации метода MDC представлена на рис.3.

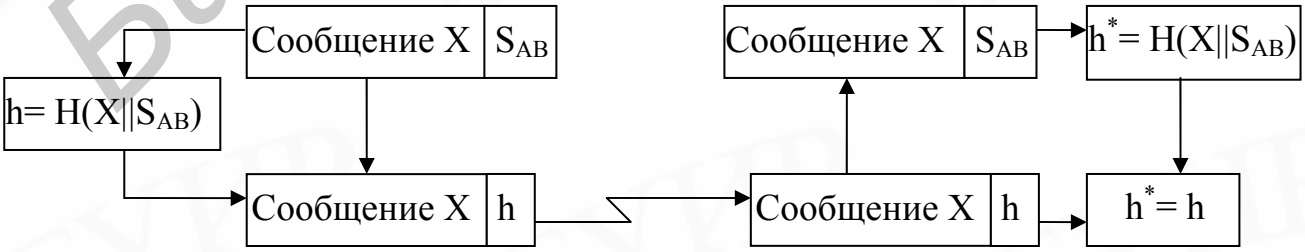

Рис.3. Реализация метода MDC без использования ключей

Соответствующая технология предполагает, что сообщающиеся стороны A и B имеют известное только им секретное значение S_{AB}. Перед отсылкой сообщения отправитель A вычисляет функцию хеширования для результата конкатенации этого секретного значения и сообщения X:

$$h = H(X \| S_{AB}).$$

Затем блок, состоящий из сообщения X и полученного хеш-кода, отправляется получателю B. Поскольку получатель B имеет значение S_{AB}, он может вычислить значение

$$h^* = H(X \| S_{AB})$$

и проверить условие $h^* = h$.

Так как само секретное значение с сообщением не пересылается, у нарушителя нет возможности модифицировать перехваченное сообщение. До тех пор пока секретное значение остается секретным, нарушитель не может навязывать ложные сообщения.

Подход, использованный в третьем варианте реализации метода MDC, принят в качестве стандарта защиты протокола IP и называется HMAC.

1.3. Метод HMAC

Функции хеширования не были разработаны для использования в качестве средства вычисления значений MAC и не могут применяться для этого непосредственно, поскольку не зависят от секретного ключа. В алгоритме HMAC реализован вариант добавления секретного ключа в уже существующие алгоритмы хеширования. Алгоритм HMAC представлен в документе RFC 2104, принят как обязательный для реализации в протоколе IPSec и используется в ряде других протоколов Internet, в частности, на транспортном уровне (прото-

кол TLS – протокол защиты транспортного уровня, призванный вскоре заменить SSL – протокол защищенных сокетов) и для электронных транзакций (протокол SET – протокол защищенных электронных транзакций).

В документе RFC 2104 представлен следующий список целей, преследовавшихся при разработке алгоритма HMAC:

- Возможность использования без модификаций уже имеющихся функций хеширования, в частности, функций хеширования, для которых существуют проверенные на практике, открытые и широко доступные программные реализации.
- Возможность легкой замены встроенной функции хеширования более скоростными или более защищенными функциями хеширования, если таковые потребуются и будут найдены.
- Сохранение скорости работы алгоритма, близкой к скорости работы соответствующей функции хеширования, без значительного ухудшения показателей скорости.
- Возможность применения ключей и простота обращения к ним.
- Простота анализа стойкости механизма аутентификации при разумных предположениях относительно используемой функции хеширования.

Последняя из вышеперечисленных целей проекта фактически обеспечивает основное преимущество HMAC по сравнению с другими схемами, основанными на использовании хеширования. HMAC обеспечивает гарантированную защищенность при условии, что встроенная функция хеширования обладает требуемой криптографической стойкостью.

Общая схема работы алгоритма HMAC представлена на рис.4.

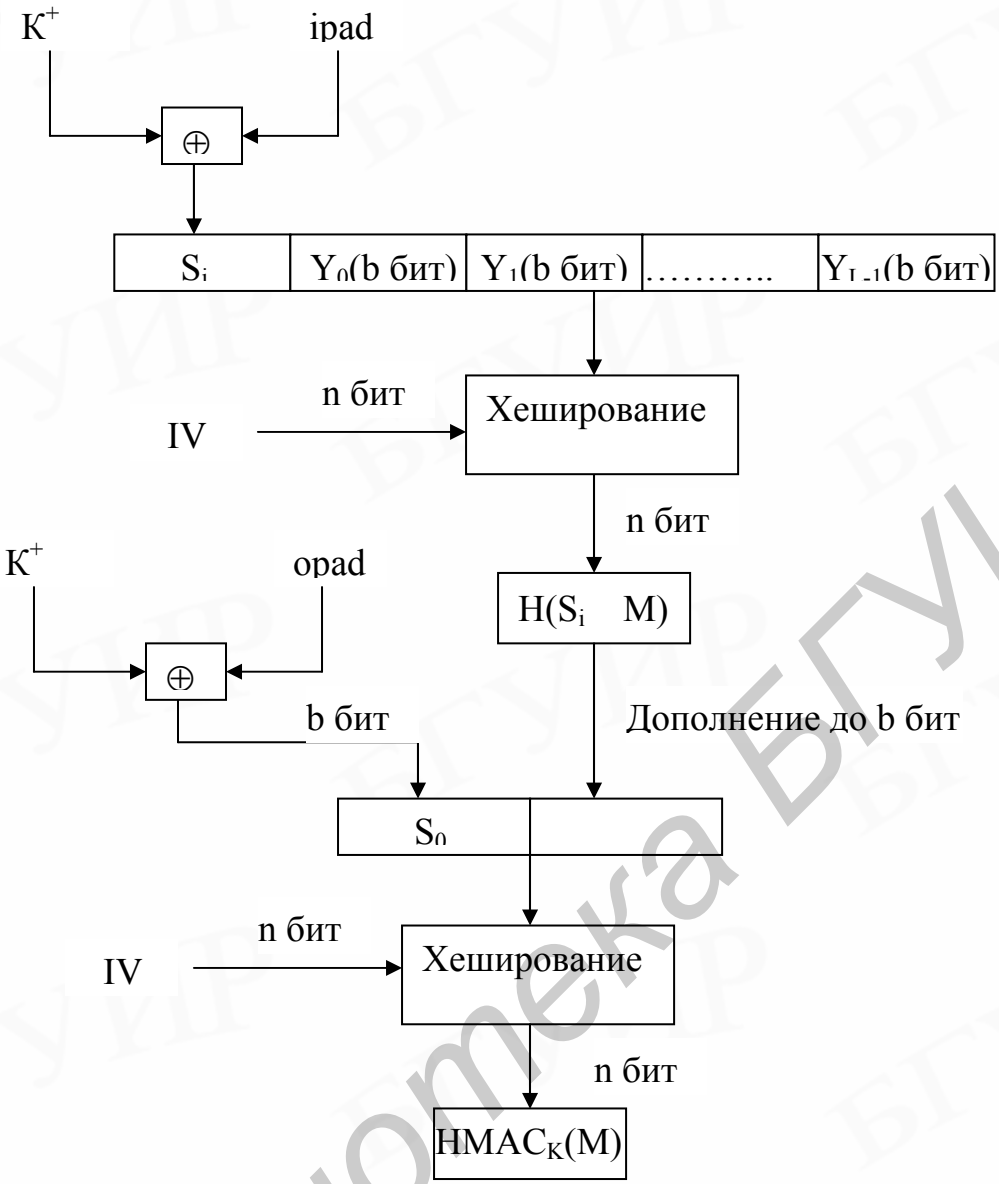

Рис.4. Схема работы алгоритма HMAC

На схеме приняты следующие обозначения:

H – встроенная функция хеширования (например SHA-1);

M – подаваемое на вход HMAC сообщение (включая биты заполнителя, требуемые встроенной функцией хеширования);

Y_i – *i*-й блок M, $0 \le i \le L - 1$;

L – число блоков в *M*;

b – число бит в блоке;

n – длина хеш-кода, генерируемого встроенной функцией хеширования;

K – секретный ключ; если длина ключа больше *b*, ключ подается на вход функции хеширования, чтобы получить *n*-битный ключ; рекомендуется длина $\geq n$;

K^+ – ключ K с добавленными в начало нулями, чтобы в результате длина получилась равной *b* битам;

ipad – значение 00110110 (шестнадцатеричное 36), повторенное b/8 раз;

opad – значение 01011010 (шестнадцатеричное 5C), повторенное b/8 раз;

IV – вектор инициализации.

Алгоритм включает в себя следующие операции:

1. К значению K слева добавляются нули, чтобы получить b-битную строку K^+ (например, если K имеет длину 160 бит и b = 512, то значение K будет дополнено 44 нулевыми байтами 0 x 00).

2. Значение K^+ связывается операцией XOR (побитовое исключающее «ИЛИ») с ipad, в результате чего получается b-битный блок S_i.

3. К S_i присоединяется M.

4. К потоку, полученному в п. 3, применяется функция H.

5. Значение K^+ связывается операцией XOR с opad, в результате чего получается b-битный блок S_o.

6. Результат хеширования, полученный в п. 4, присоединяется к S_o.

7. К потоку, полученному в п. 6, применяется функция H, и результат подается на выход.

Алгоритм HMAC можно представить формулой

$$\mathrm{HMAC}_K = \mathrm{H}[(K^+ \oplus \mathrm{opad}) \,\|\, \mathrm{H}[(K^+ \oplus \mathrm{ipad}) \,\|\, M]].$$

Для достаточно длинных сообщений алгоритм HMAC должен выполняться приблизительно за время, необходимое встроенной функции хеширования.

В случае метода MAC использование секретного ключа при вычислении проверочной комбинации дает следующее ограничение на длину проверочной комбинации: $n \geq 32$.

В случае метода MDC на длину n проверочной комбинации накладывается условие $n \geq 128$. Это вызвано необходимостью обеспечения защиты от метода криптоанализа под названием «день рождения».

2. ИМИТОЗАЩИТА ШИРОКОВЕЩАТЕЛЬНЫХ СООБЩЕНИЙ

В современных телекоммуникационных сетях имеется возможность одновременной передачи одного и того же сообщения целой группе получателей. Это так называемый режим широковещательного вида доведения сообщений, который, как правило, предоставляется отдельным пользователям системы. В этом случае задача обеспечения имитозащиты информации имеет специфические особенности, связанные с обеспечением предотвращения выдачи циркулярных сообщений пользователями, не имеющими на это право, но одновременно являющимися получателями этих сообщений.

За рубежом наибольшее распространение получила схема аутентификации циркулярных сообщений, разработанная американским агентством DARPA (Defence Advaced Research Projects Agency). Суть схемы DARPA состоит в том, что каждая пара пользователей, например A и B, имеет парный ключ взаимодействия K_{AB}. Каждый такой ключ известен только паре пользователей и, возможно, центру распределения ключей.

Все сообщения, подлежащие аутентификации, обрабатываются блочным шифром DES в режиме CBC (шифрование со сцеплением блоков). Сообщение, которое пользователем A передается пользователям B и C, защищается таким образом, чтобы оба получателя смогли проверить подлинность источника сообщения и содержания сообщения. Для этого пользователь A выполняет следующие процедуры:

1. Генерирует случайный ключ аутентификации данных К (DAK – Data Autohentication Key).

Этот ключ используется для защиты одного, и только одного, сообщения.

2. В режиме шифрования ECB (электронная кодовая книга – простая замена) вырабатываются два шифротекста:

$$\overline{K}_{AB} = E_{K_{AB}}(K) \text{ и } \overline{K}_{AC} = E_{K_{AC}}(K).$$

3. Сообщение М зашифровывается в режиме CBC на ключе К (DAK) с вектором инициализации (синхропосылкой), состоящим из нулей. Все блоки шифротекста, за исключением последнего, отбрасываются. Последний блок используется в качестве контрольной комбинации Z (MAC): $Z = MAC_k(M)$.

4. В режиме шифрования ECB вырабатываются два шифротекста:

$$Z_{AB} = E_{K_{AB}}(Z) \text{ и } Z_{AC} = E_{K_{AC}}(Z).$$

5. Пользователям В и С посылается защищенное сообщение в виде M, $\overline{K}_{AB}$, Z_{AB}, $\overline{K}_{AC}$, Z_{AC}.

По схеме DARPA пользователь В (аналогично и С), получив сообщение M^*, $\overline{K}_{AB^*}$, Z_{AB^*}, $\overline{K}^*_{AC}$, Z^*_{AC} (возможно, искаженное), выполняет следующие процедуры:

1. Расшифровывает DAK: $K^* = D_{K_{AB}}(\overline{K}^*_{AB})$.

2. Вычисляет MAC: $Z^* = MAC_K^*(M^*)$.

3. Зашифровывает Z^* на ключе K_{AB}: $Z_{AB}^{**} = E_{K_{AB}}(Z^*)$.

4. Результат этого зашифрования Z_{AB}^{**} сравнивает с полученной в составе сообщения комбинацией Z_{AB}^*.

В случае совпадения полученное сообщение считается подлинным:

$$M^* = M,\ K^*_{AB} = \overline{K}_{AB},\ Z^*_{AB} = Z_{AB},\ K^* = K.$$

Однако, как показывает анализ, данная схема при определенных условиях может оказаться несостоятельной. Существует стратегия, пользуясь которой, злоумышленник – пользователь С может послать пользователю В новое ложное циркулярное сообщение от имени пользователя А. Эта стратегия заключается в том, что пользователь С, сформировав ложное сообщение L (затем оно будет дополнено еще одним блоком), выполняет следующие процедуры:

1. Выбирает контрольную комбинацию Z от некоторого подлинного ранее переданного пользователем А циркулярного сообщения М.

2. Выбирает ключ К (DAK) для этого сообщения.

3. Зашифровывает Z на ключе К: $x = E_k(Z)$.

4. В режиме шифрования CBC вычисляет $y = MAC_k(L)$.

5. Вычисляет $m = x + y$, где «+» – сложение по модулю два 64-битных блоков.

6. Дополняет L блоком m и получает ложное сообщение (Faulse Message): FM = L,m.

7. Посылает пользователю В ложное сообщение от имени пользователя А в виде FM, K_{AB}, Z_{AB}, $\overline{K}_{AC}$, Z_{AC}.

Пользователь В, получив сообщение, выполняет следующие процедуры:

1. Восстанавливает ключ К (DAK): $K = D_{K_{AB}}(K_{AB})$.

2. Вычисляет $Z = MAC_k(FM)$ (последовательно вычисляются $y = MAC_k(L)$, $x = y + m$, $z = E_k(x)$).

3. Зашифровывает Z на ключе K_{AB} : $Z_{AB} = E_{K_{AB}}(Z)$.

4. Сравнивает вычисленное значение Z_{AB} с полученным значением Z_{AB}, обнаруживает совпадение и делает вывод о подлинности полученного сообщения.

Таким образом, пользователь С навязывает пользователю В ложное циркулярное сообщение от имени пользователя А.

Основная причина возможности осуществления такой стратегии навязывания ложных сообщений состоит в том, что исходная контрольная комбинация Z и для B, и для C вычисляется на одном ключе K, что в совокупности с особенностями режима шифрования CBC приводит к несостоятельности схемы DARPA.

Возможны два подхода к решению данной проблемы. Первый подход подразумевает использование различных функций (в том числе и необратимых) для вычисления контрольной комбинации (функциональная независимость). Второй подход подразумевает, что для каждого получателя контрольная комбинация вычисляется совершенно независимо (кодовая независимость).

2.1. Метод функциональной независимости контрольных комбинаций

Суть метода заключается в следующем. Отправитель сообщения вычисляет контрольную комбинацию Z при помощи функции сжатия (хеш-функции): Z = F(M). Затем зашифровывает Z на парных ключах взаимодействия с получателями (пусть их число r) и все r зашифрованных комбинаций посылает вместе с сообщением.

Эффективность этого метода основана на свойствах функции сжатия (хеш-функции) и на шифровании контрольных комбинаций.

В схеме DARPA функция сжатия реализована на основе режима шифрования CBC алгоритма DES с ключом, известным всем законным получателям сообщения M. Такая функция не является необратимой.

Даже в случае наилучшего выбора функции F длина Z должна быть достаточной для того, чтобы защититься от метода криптоанализа типа «день рождения», значительная вероятность успеха при котором достигается при реализации вычислений объемом порядка $\sqrt{2^n}$ операций, где n – длина в битах. Для этого выбирают n = 128.

Однако существует возможность защититься от метода криптоанализа «день рождения» не путем увеличения длины, а путем добавления в качестве первого – блока из случайных бит, что также позволяет исключить возможность заблаговременной заготовки пар подлинных и ложных сообщений (M_i, FM_i), i = 1,2…, таких, что $F(M_i) = F(FM_i)$. При таком подходе можно брать n = 64.

Функция шифрования контрольной комбинации Z применяется, чтобы исключить возможность подмены значения Z. Шифрование производится на парных ключах взаимодействия отправителя A с r получателем сообщения M:

$$Z_{Ai} = E_{K_{Ai}}(Z),\ i=1,..,r\ .$$

Действительно, если ключ данного шифрования известен только двоим – отправителю и получателю, то возможность подмены исключена. Таким образом, защищенное сообщение передается в следующем виде:

$$M, Z_{A_1}, Z_{A_2}, ..., Z_{A_r}\ .$$

Более удачной является функция, реализуемая по схеме, описанной ниже. В этой схеме сообщение M последовательно разбивается на блоки по 56 бит: M_1, M_2, …, M_n. Затем, используя фиксированное значение C_0, производят итеративное вычисление 64-битовых блоков:

$$C_i = C_{i-1} + E_{Mi}(C_{i-1}),$$

где под $E_{Mi}(C_{i-1})$ понимается результат шифрования блока C_{i-1} на ключе M_i при помощи блочного криптоалгоритма («+» означает сложение по модулю два 64-битных блоков).

В качестве контрольной комбинации выбирается последний блок C_n:

$$Z = C_n.$$

На основе свойств криптоалгоритма DES доказана необратимость функции, реализованной на такой схеме. Однако не рассмотрена возможность осуществления метода криптоанализа «день рождения». Следовательно, необходима модификация схемы DMW: либо расширять Z до 128 бит (двойной счет с различными C_0), либо в качестве первого блока добавлять к сообщению блок из случайных бит. Последнее решение более эффективно.

Таким образом, можно заключить, что для реализации схемы аутентификации циркулярных сообщений необходима реализация необратимой функции и блочного шифра. Для реализации необратимой функции может быть использована схема DMW при условии добавления к сообщению блока случайных бит.

2.2. Метод кодовой независимости контрольных комбинаций

Альтернативный подход подразумевает независимое вычисление отправителем A контрольных комбинаций для каждого из r получателей на основе алгоритма DES в режиме CBC (блочного криптоалгоритма в режиме со сцеплением блоков) с использованием сеансовых ключей $K_1, K_2, \ldots, K_r$:

$$Z_i = MAC_{Ki}(M), \qquad i = 1, \ldots, r.$$

Сеансовые ключи шифруются в режиме простой замены (режим ECB) с использованием парных ключей взаимодействия:

$$\overline{K}_{A_1} = E_{K_{A_1}}(K_1), \overline{K}_{A_2} = E_{KA_2}(K_2), \ldots.$$

Таким образом, для передачи циркулярного сообщения M к нему добавляется r пар:

$$(Z_1, \overline{K}_{A_1}), (Z_2, \overline{K}_{A_2}), ..., (Z_r, \overline{K}_{A_r}).$$

При этом пара $(Z_i, \overline{K}_{A_i})$ предназначена для получателя с номером i. Возможность осуществления злоумышленником навязывания ложного циркулярного сообщения исключена.

Возможно вычисление контрольных комбинаций с использованием парных ключей взаимодействия:

$$Z_i = MAC_{K_{Ai}}(M),\ i=1,...,r.$$

В этом случае для передачи циркулярного сообщения M к нему добавляются комбинации Z_1', Z_2',…, Z_i', где под Z_i понимается блок из 32 первых бит блока Z_i.

Очевидно, что этот второй альтернативный метод более прост. Однако следует заметить, что предыдущий метод требует меньше вычислений, так как он подразумевает лишь одноразовое сжатие сообщения M ($Z = F(M)$), второй метод подразумевает r подобных операций $Z_i = E_{Ki}(M)$, $i = 1,…, r$.

Оба метода реализуются с помощью стандартного криптографического алгоритма типа DES или ГОСТ 28147-89.

Таким образом, имитозащита информации для циркулярных вариантов передачи в принципе может быть обеспечена с помощью симметричных схем аутентификации сообщений. При этом в каждое циркулярное сообщение для осуществления аутентификации вводится дополнительная избыточность, объем которой пропорционален числу абонентов в циркуляре.

Контрольные вопросы и задания

1. Опишите модель имитозащиты сообщений. От каких способов нарушения подлинности сообщений она используется?

2. Разработайте схему реализации имитозащиты на основе метода MAC с использованием алгоритма DES.

3. Чему равна вероятность навязывания ложного сообщения, если длина MAC составляет 48 бит?

4. Приведите варианты схем реализации метода MDC.

5. В чём отличие методов функциональной и кодовой независимости формирования контрольной комбинации при имитозащите широковещательных сообщений?

ЛИТЕРАТУРА

1. Бобов М.Н. , Конопелько В.К. Обеспечение безопасности информации в телекоммуникационных системах. – Мн. : БГУИР, 2002.–164 с.

2. Иванов М.А. Криптографические методы защиты информации в компьютерных системах и сетях.– М.: КУДИЦ - ОБРАЗ, 2001.– 368 с.

3. 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.

4. Симонс Г.Дж. Обзор методов аутентификации информации //ТИИЭИР.–1988.– Т.76. – № 5.– С.105 – 125.

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

6. Уолрэнд Дж. Телекоммуникационные и компьютерные сети. Вводный курс.– М.: Постмаркет, 2001.

Св. план 2005, поз.97

Учебное издание

Бобов Михаил Никитич

ИМИТОЗАЩИТА СООБЩЕНИЙ В СЕТЯХ СВЯЗИ

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 Т.А. Лейко
Корректор Н.В. Гриневич

Подписано в печать 18.01.2005. Формат 60x84 1/16. Бумага офсетная.
Гарнитура «Таймс». Печать ризографическая. Усл. печ. л. 1,51.
Уч.-изд. л. 1,1. Тираж 100 экз. Заказ 645.

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