КРИПТОГРАФИЧЕСКИЕ ПРОТОКОЛЫ В СИСТЕМАХ С ОГРАНИЧЕННЫМИ РЕСУРСАМИ

Р. В. Мещеряков, С. К. Росошек, А. А. Шелупанов
Томский государственный университет систем управления и радиоэлектроники, Россия
e-mail: mrv@security.tomsk.ru, office@keva.tusur.ru

М. А. Сонькин
Отдел проблем информатизации ТНЦ СО РАН, Томск, Россия
e-mail: sonkin@cc.tpu.edu.ru

Cryptographic protocols used for data transmission in a system with limited resources are considered in the case of the distributed organization of information processing for various operating modes. Theoretical foundations for the establishment of a uniform key and the establishment of a session of communication are presented.

В ряде вычислительных телекоммуникационных систем требуется производить обмен конфиденциальной информацией. Очевидно, что полноценный защищенный обмен может быть реализован с помощью криптографических протоколов. Криптографическим протоколом называется протокол, в основе которого лежит криптографический алгоритм.

Практически каждая телекоммуникационная система требует, чтобы в начале сеанса работы пользователь идентифицировал себя. Обычно пользователю предлагается ввести имя и пароль. Пользователь помнит этот секрет, а сервер хранит либо копию секрета, либо значение, вычисленное на основе секрета. Во время аутентификации происходит сопоставление пароля, введенного пользователем, и значения, хранимого сервером. Аутентификация при помощи паролей — наиболее распространенный вид аутентификации. Если злоумышленник знает чужой пароль, то имеет возможность выдавать себя за другого субъекта и сервер не может отличить его от настоящего пользователя.

Существует несколько способов получения секретного пароля в сети [3]. Пользователь Z (злоумышленник) может использовать программу-анализатор, или сниффер. Атаки анализаторов обнажают две серьезные проблемы аутентификации при помощи паролей. Во-первых, для аутентификации пользователь A должен передать свой пароль, разделенный секрет. Выполняя это, пользователь A может раскрыть его. Во-вторых, если разделенный секрет пользователя A используется долгое время, пользователю C достаточно получить пароль один раз, после чего он может выдавать себя

за пользователя A, пока последний не изменит свой пароль. Эти слабые стороны делают атаки анализаторов успешными. Эволюция механизмов аутентификации началась в ответ на атаки анализаторов, появилась идентификация односторонняя и взаимная. Классификация систем представлена на рис. 1.

На первом этапе работы практически любой телекоммуникационной системы необходимо использовать идентификацию. Представим телекоммуникационную систему в виде рис. 2.

С другой стороны, в качестве оконечного оборудования сети передачи данных Ai могут быть абонентские пункты, не обладающие достаточными ресурсами для реализации сложных криптографических протоколов. Таким образом, ставится задача рассмотреть

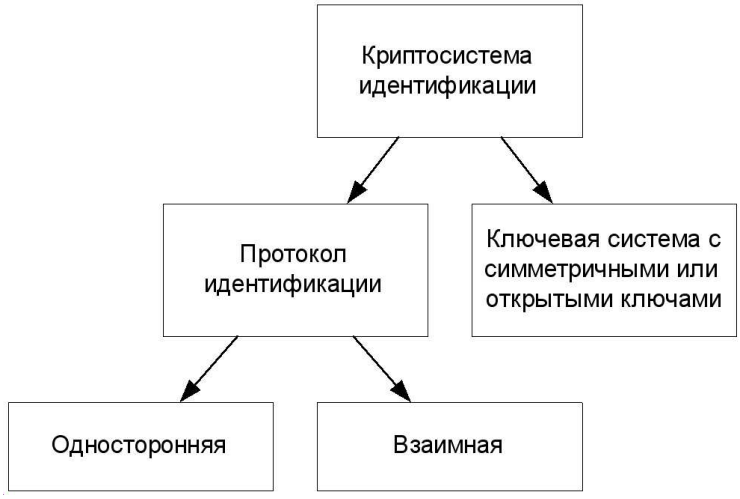

Рис. 1. Схема криптосистемы идентификации

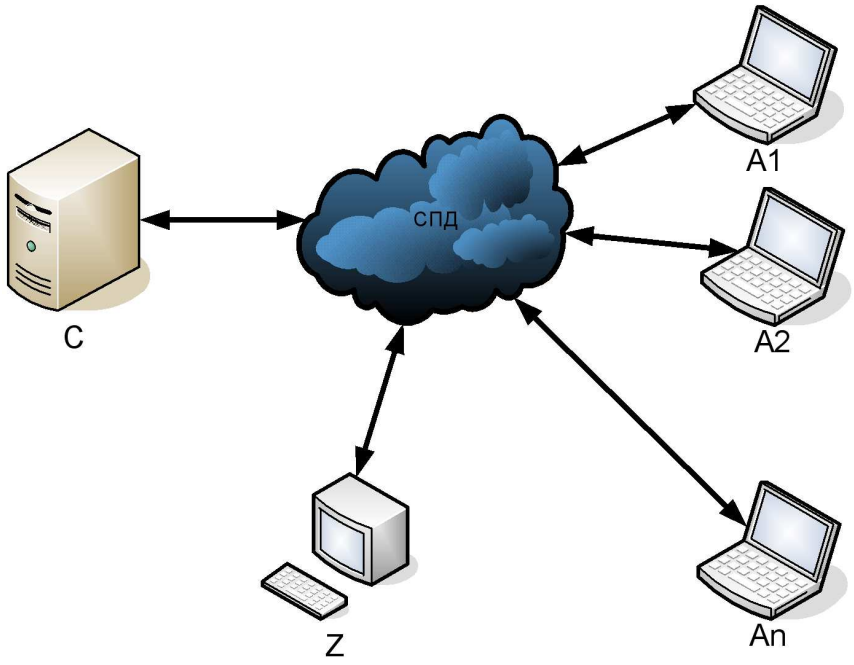

Рис. 2. Сеть передачи данных

криптографические системы при использовании систем с ограниченными вычислительными ресурсами.

В качестве экспериментальной сети передачи данных была определена сеть на основе пакетного контроллера "ВИП-М" (производитель ООО "Инком"), позволяющего в пакетном режиме передавать цифровую информацию посредством использования существующих сетей передачи данных. Основные концепции по встраиванию криптографических функций в ВИП изложены в [1].

1. Теоретические аспекты применения протоколов

Механизм односторонней идентификации с временным штампом. Будем считать, что A аутентифицируется в системе у пользователя B. Для описания алгоритмов идентификации введем обозначения: R_A — случайное число, генерированное A; t_A — временный штамп; S_A — механизм цифровой подписи A; cert_A — сертификат открытого ключа A для S_A; $A \to B : \mathrm{cert}_A, t_A, B, S_A(t_A, B)$.

После полученного сообщения пользователь B проверяет подлинность временного штампа, подлинность полученного идентификатора B и (используя открытый ключ A, выделенный из cert_A, после его проверки) проверяет, что подпись $S_A(t_A, B)$ корректна. Схема протокола представлена на рис. 3.

Механизм односторонней идентификации со случайными числами. Временный штамп может быть заменен на случайное число ценой дополнительного сообщения:

$$A \leftarrow B : R_B,$$

$$A \to B : \mathrm{cert}_A, R_A, B, S_A(R_A, R_B, B).$$

Пользователь B проверяет подлинность своего идентификатора и, используя открытый ключ подписи A из cert_A, проверяет подлинность подписи $S_A(R_A, R_B, B)$. Подписание R_A предотвращает атаку с выбранным текстом (рис. 4).

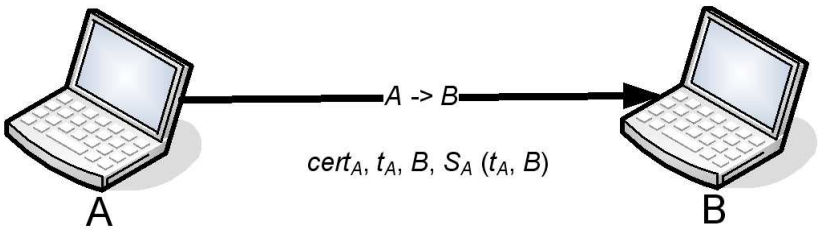

Рис. 3. Протокол односторонней идентификации с временным штампом

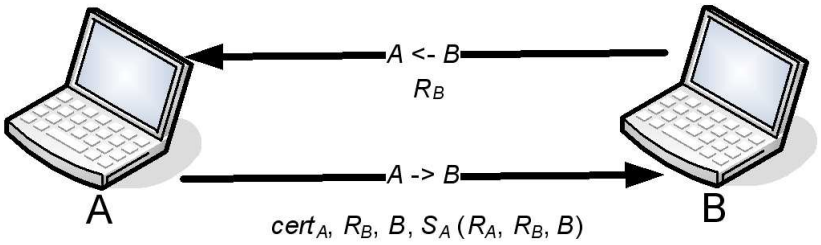

Рис. 4. Протокол односторонней идентификации со случайными числами

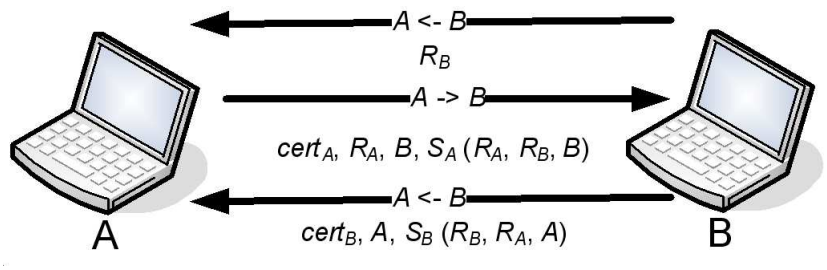

Рис. 5. Протокол взаимной идентификации со случайными числами

Поскольку запрос генерируется случайным образом, пользователь Z не может повторно использовать подпись, сгенерированную пользователем A, чтобы выдать себя за него. Значение, которое отправляет пользователь A, аутентифицирует его идентичность только один раз. Имя пользователя A передается открыто, и нет причин его скрывать.

Механизм взаимной идентификации со случайными числами. Чтобы этот механизм был пригоден для взаимной аутентификации, необходимы еще один запрос и ответ (рис. 5):

$$A \leftarrow B : R_B,$$

$$A \rightarrow B : \mathrm{cert}_A, R_A, B, S_A\left(R_A, R_B, B\right),$$

$$A \leftarrow B : \mathrm{cert}_B, A, S_B\left(R_B, R_A, A\right).$$

Итак, механизмы идентификации при помощи сертификатов поддерживают идентификацию в открытой сети (на многих удаленных серверах) и обеспечивают взаимную аутентификацию, а также не требуют активного участия третьих сторон. Для успешной аутентификации должны быть доступны только пользователь и сервер/пользователь.

2. Протоколы аутентификации Шнорра и Фейге — Фиата — Шамира

Одним из наиболее эффективных практических протоколов аутентификации является протокол Шнорра [12].

Пусть p и q — простые числа, причем q делит $p-1$. Пусть $g \in GF(p)$, где $GF(p)$ — конечное кольцо классов вычетов, такое, что $g^q \equiv 1 (\mathrm{mod}\, p)$, $g \neq 1$. Далее, пусть $k \in GF(q)$ и $y = g^{-k} (\mathrm{mod}\, p)$.

Таким образом, опять возникла задача дискретного логарифмирования, т. е. по заданному значению y и при известных p, q и g необходимо найти k. Тогда алгоритм аутентификации будет состоять в следующем:

— абонент A выбирает случайное число a из множества $\{1, \ldots, q-1\}$, вычисляет $r = g^a (\mathrm{mod}\, p)$ и посылает его абоненту B (величина r может быть вычислена заранее);

— абонент B выбирает случайное число e из множества $\{1, \ldots, 2^{t-1}\}$, где t — некоторый параметр, и посылает его абоненту A;

— абонент A вычисляет $s = a + ke (\mathrm{mod}\, q)$ и посылает его абоненту B;

— абонент B проверяет соотношение $r = g^s y^e (\mathrm{mod}\, p)$ и, если оно выполняется, принимает доказательство, иначе отвергает.

Т а б л и ц а 1. Протокол аутентификации Файге — Фиата — Шамира

Предварительный этап		
A	Центр доверия T	B
$\lambda = (\lambda_1, \lambda_2, \ldots, \lambda_k)$ — открыт. ключ $\beta = (\beta_1, \beta_2, \ldots, \beta_k)$ — закрыт. ключ	$n = pq$ — случ. $(\lambda_1, \lambda_2, \ldots, \lambda_k), \lambda_I \in QR_n$ $(\beta_1, \beta_2, \ldots, \beta_k) : \beta_i = \min\{\mathrm{sqrt}(\lambda I^{-1}) \bmod n\}$	
$n, \lambda_1, \lambda_2, \ldots, \lambda_k$		

Рабочий этап			
	A		B
for $(i = 1, 2, \ldots, t)$			
1	a_i — случ., $a_i < n$, $r_i = a_i^2 \bmod n$	$\rightarrow$	
2		$\leftarrow$	$(c_{i1}, c_{i2}, \ldots, c_{ik}) \in \{0,1\}^k$ — случ.
3	$y_i = a_i \left(\beta_1^{c_{i1}} \beta_2^{c_{i2}} \ldots \beta_k^{c_{ik}}\right) (\mathrm{mod}\, n)$	$\rightarrow$	
4			$r_i = y_i^2((1c_{i1}.(2c_{i2}. \ldots .(kc_{ik})(\mathrm{mod}\, n)$

Существуют также и другие схемы. Рассмотрим из них схему аутентификации Фейге — Фиата — Шамира, которая показана в табл. 1.

Пусть n — произведение двух больших простых чисел. Для генерации открытых и закрытых ключей абонент A выбирает k различных чисел λ_1, λ_2, ..., λ_k, каждое из которых является квадратичным вычетом по модулю n. Строка λ_1, λ_2, ..., λ_k служит открытым ключом. Затем вычисляются наименьшие значения β_1, β_2, ..., β_k, для которых $\beta_i = \mathrm{sqrt}\left(\lambda_I^{-1}\right) (\mathrm{mod}\, n)$. Строка β_1, β_2, ..., β_k служит секретным ключом. Далее выполняется следующий протокол:

— абонент A выбирает случайное число a из множества $\{1, \ldots, n-1\}$, вычисляет $r = a^2 (\mathrm{mod}\, n)$ и посылает его абоненту B;

— абонент B посылает A строку из k случайных битов — c_1, c_2, ..., c_k;

— абонент A вычисляет $y = a\left(\beta_1^{c_1} \beta_2^{c_2} \ldots \beta_k^{c_k}\right) (\mathrm{mod}\, n)$ и посылает его абоненту B;

— абонент B проверяет, что $r = y^2 \left(\lambda_1^{c_1} \lambda_2^{c_2} \ldots \lambda_k^{c_k}\right) (\mathrm{mod}\, n)$.

Абоненты A и B повторяют этот протокол t раз, пока B не убедится, что A знает β_1, β_2, ..., β_k. Шанс, что A обманет B t раз, равен 1 из 2^{kt}.

3. Протоколы аутентификации с нулевым разглашением

Протоколы аутентификации, обладающие доказательством с нулевым разглашением (ZK-протоколы), позволяют произвести процедуры идентификации, обмена ключами и другие основные криптографические операции без утечки любой секретной информации в течение информационного обмена [4]. Этой цели можно добиться при помощи демонстрации знания секрета, однако проверяющий должен быть лишен возможности получать дополнительную информацию о секрете. Иными словами, ZK-протоколы позволяют установить истинность утверждения, не передавая какой-либо дополнительной информации о самом утверждении. ZK-протоколы являются системой интерактивного доказательства, в которой проверяющий и доказывающий обмениваются многочисленными запросами и ответами. Целью доказывающего является убеждение проверяющего в истинности утверждения. Поверяющий отклоняет или принимает доказательство. Та-

ким образом, ZK-протоколы носят вероятностный, а не абсолютный характер. Сторона A владеет секретом s и пытается убедить сторону B в знании секрета. Стоит отметить, что доказательство знания секрета отличается от доказательства того факта, что секрет существует.

Основные характеристики ZK-протокола:

— проверяющий не может ничего узнать из протокола;

— доказывающая сторона не может обмануть проверяющую сторону.

Если сторона A не знает секрета s и пытается доказать стороне B его знание, то после нескольких раундов протокола данный факт может быть установлен настолько точно, насколько это необходимо. Протокол также является "Cut AND Choose", т. е. после первого неудачного раунда сторона B точно знает, что A нелегальна. Проверяющая сторона не может обмануть доказывающую сторону. Сторона B не может вынести из протокола какой-либо информации, даже если она не следует протоколу. Единственное, что может сделать сторона B, это убедить себя, что сторона A знает секрет. Доказывающая сторона всегда раскрывает только одно из многих решений любой поставленной проблемы и никогда все, что позволило бы найти сам секрет. Проверяющая сторона не может стать доказывающей для любой третьей стороны.

Существует три основных режима работы ZK-протокола:

— интерактивный, когда стороны A и B интерактивно взаимодействуют через протокол, шаг за шагом проверяя достоверность;

— параллельный, когда сторона A формулирует ряд запросов, а сторона B в это время запрашивает ряд ответов на поставленные задачи. Данный режим может быть использован, чтобы снизить количество интерактивно передаваемых сообщений при плохом уровне связи;

— автономный, когда сторона A формулирует ряд запросов, затем, играя роль стороны B, использует криптографически стойкую однонаправленную хеш-функцию набора, чтобы выбрать произвольные решения для каждого запроса. После этого A добавляет ряд полученных решений к сообщению. Этот режим может быть использован для цифровой подписи.

Один из наиболее известных ZK-протоколов идентификации — это протокол Фиата — Шамира. Схема его работы представлена в табл. 2.

Сторона A доказывает B знание секрета s за t раундов по три сообщения в каждом.

Параметры протокола:

— доверительный центр T выбирает и публикует модуль n (512–1024 бита), являющийся произведением двух больших чисел ($n = pq$, p и q сохраняются в секрете);

— каждый доказывающий выбирает секрет s, взаимно простой с n, $1 \le s \le n-1$, вычисляет $v = s^2 \bmod n$ и регистрирует v у T в качестве своего открытого ключа.

Сообщения, передаваемые в рамках каждого раунда:

$$A \to B : x = r^2 \bmod n;$$

$$A \leftarrow B : e \in \{0, 1\};$$

$$A \to B : y = rs^e \bmod n.$$

B принимает доказательства за t раундов, и последовательность действий в рамках протокола имеет вид:

— A выбирает случайное число r, $1 \le r \le n-1$, и посылает B $x = r^2 \bmod n$;

Таблица 2. Протокол Фиата — Шамира

Предварительный этап			
A	Центр доверия T		B
β, λ	$n = pq$ — случ. $\lambda \in QR_n$ $\beta = \min\{\text{sqrt}\,(\lambda^{-1}) \bmod n\}$		
Рабочий этап			
	A		B
1	a — случ., $a < n$ $r = a^2 \bmod n$	$\rightarrow$	
2		$\leftarrow$	$e \in \{0, 1\}$ — случ.
3	if $(e = 0)$ r if $(e = 1)$ $y = a\beta (\bmod n)$	$\rightarrow$	
4			if $(e = 0)$ $r =?$ $a^2 (\bmod n)$ (Знает ли A $\text{sqrt}(r)$?) if $(e = 1)$ $r =?$ $y^2\lambda (\bmod n)$ (Знает ли A $\text{sqrt}(\lambda^{-1})$?)

— B выбирает случайным образом e и посылает его A;

— A вычисляет y и посылает его B, где $y = r(e = 0)$ или $y = rs(e = 1)$;

— B отвергает доказательство, если $y = 0$, иначе производится проверка $y^2 \equiv xv^e \bmod n$.

В зависимости от e $y^2 = x^2 \bmod n$ или $y^2 = xv \bmod n$, иначе $v = s^2 \bmod n$. Число раундов выбирается от 20 до 40.

Протокол может быть разработан с помощью идентификации, основанной на RSA публичном ключе и комбинированной со свойствами ZK-протоколов, путем обмена ключами. Сторона B может зашифровать произвольное число с помощью публичного ключа стороны A, и если A может расшифровать его с помощью своего секретного ключа, то A идентифицировано. Чтобы скрыть возвращаемое зашифрованное значение, используется однонаправленная хеш-функция. Таким образом, B может только проверить, что присланное стороной A значение соответствует хеш-функции от выбранного числа. Сессионный ключ может быть помещен в некоторые биты произвольного числа, которые никогда не пересылаются в чистом виде.

Большинство ZK-протоколов может быть использовано для цифровых подписей, если сторону B заменить криптографически стойкой однонаправленной хеш-функцией. Сторона A может сформулировать ряд запросов, использовать однонаправленную хеш-функцию как виртуальную строну B (которая произвольным образом потребует один ответ на каждый запрос) и предоставить эти ответы. В качестве аргументов хеш-функции используются набор ответов и запросов. Таким образом, ни ответ, ни запрос не может быть изменен без изменения подписи. Результат действия “хорошей” криптографической однонаправленной хеш-функции является полностью произвольным и непредсказуемым. Принимающая сторона вычисляет значение хеш-функции и проверяет корректность ответов на запросы. Если проверка пройдена, то подпись может считаться верной. Рассмотрим в качестве примера схему ЭЦП Шнорра, она представлена в табл. 3.

Таблица 3. Протокол аутентификации Шнорра

Предварительный этап		
A	Центр доверия T	B
$s \in_R \{1, \ldots, q\}$ $v = a^{-s} \bmod p$	p, q — простые числа, $q \mid p-1$, $a \in Z_p : a^q \equiv 1 (\bmod p)$	
p, q, a		

Рабочий этап			
	A		B
1	$r \in \{1, \ldots, q-1\}$, $x = a^r \bmod p$ — случ.	$\rightarrow$	
2		$\leftarrow$	$e \in \{0, \ldots, 2^{t-1}\}$ — случ.
3	$y = r + se \bmod q$	$\rightarrow$	
4			$x \stackrel{?}{=} a^y v^e \bmod p$

Пусть p и q — простые числа, такие что p делит $q-1$, пусть g принадлежит Zp, $g^q = 1 \bmod p$, g не равно 1. В качестве секретного ключа выбирается x, принадлежащий $\{1, \ldots, q-1\}$. Открытый ключ $y = g^{(-x)} \bmod p$.

— A выбирает случайное число k, принадлежащее $\{1, \ldots, q-1\}$, и вычисляет $r = g^k \bmod p$;

— A вычисляет $e = h(r, m)$, где m — подписываемое сообщение;

— A вычисляет $s = (k + ex) \bmod q$ и посылает сообщение m с подписью (e, s) получателю B.

B вычисляет $r1 = g^s y^e \bmod p$ и проверяет, выполняется ли равенство $e = h(r1, m)$. Если да, то подпись принимается, в противном случае — отвергается. Преимущество схемы Шнорра перед схемой Эль Гамаля заключается в том, что k выбирается из меньшего множества (длина k порядка 140 битов). Это повышает эффективность вычисления дискретных экспонент. Кроме того, стоит заметить, что использование в схеме Шнорра хеш-функции при вычислении e и приведение подписи s по модулю q сокращают длину подписи по сравнению со схемой Эль Гамаля. Длина подписи — один из важнейших показателей эффективности схемы.

Далее приведена сравнительная таблица системных требований для различных семейств протоколов (табл. 4).

Из приведенной таблицы видно, что ZK-протоколы могут использоваться в системах с повышенным требованием к безопасности, но их применение предъявляет жесткие требования к вычислительным способностям и размеру памяти. Это накладывает ряд ограничений в телекоммуникационной системе.

Таблица 4. Системные требования для различных протоколов

Семейство протоколов	Размер сообщений	Количество раундов	Количество вычислений	Требования к памяти
Zero-knowledge	Большой	Много	Большое	Большая
Public-key	Большой	Один	Очень большое	Большая
Symmetric	Маленький	Один	Маленькое	Маленькая

4. Практическая реализация криптографических протоколов

Реализация криптографических протоколов должна проводиться в соответствии с действующим законодательством. Использование электронно-цифровой подписи осуществляется согласно ГОСТ 34.10-2001, реализация которого требует существенно бóльших вычислительных ресурсов по сравнению с режимом симметричного шифрования. Для реализации необходимо использовать различные режимы ГОСТ 28147-89, в том числе с модификациями. Очевидно, что сложностей с реализацией шифрования нет, наибольшее внимание должно быть уделено процедурам идентификации и установления общего ключа.

Отметим, что в базовой схеме работы пакетного контроллера существует два режима работы:

— связь "точка — точка", позволяющая обеспечивать обмен информацией двух абонентов системы при наличии установившегося канала передачи данных;

— связь "широковещательное оповещение", позволяющая передавать информацию от одного абонента одномоментно нескольким абонентам системы.

В каждом из режимов определен инициатор связи, называемый центром и абонентом системы (сторона A). При различных реализациях криптографических протоколов может быть реализовано несколько режимов работы защищенной сети на основе пакетного контроллера.

Криптопроцессор имеет встроенный датчик случайных (псевдослучайных) чисел — ДСЧ (ДПСЧ). Сеанс связи "точка — точка".

В центре имеется библиотека чисел p и α, которые определяют основные параметры выработки общего ключа. Так как размер библиотеки значительно ограничен, используется расширение числа p посредством имитовставки $p' = h(p)$, что позволяет увеличить длину ключа. Затем с помощью ДСЧ (ДСПЧ) выбирают случайное целое число x в интервале $1 < x < p$ и вычисляют $\gamma = \alpha^x \bmod p$. Соответственно, для γ также применяется функция имитовставки $\gamma' = h(\gamma)$. Получившаяся тройка $\langle p', \gamma', \alpha \rangle$ передается по открытому каналу.

Абонент после получения тройки $\langle p', \gamma', \alpha \rangle$ убирает имитовставки и получает тройку $\langle p, \gamma, \alpha \rangle$. Затем абонент выбирает случайное целое y в интервале $1 < y < p$, вычисляет $\mu = \alpha^y \bmod p$, делает имитовставку для μ и полученное число μ' отправляет в центр. Затем абонент вычисляет общий с центром ключ $K = \gamma^y \bmod p = \alpha^{xy} \bmod p$. С другой стороны, центр, получив μ' от абонента, убирает имитовставку и получает μ. Затем вычисляет общий с абонентом ключ: $K = \mu^x \bmod p = \alpha^{yx} \bmod p$.

Центр шифрует ключом K посредством ГОСТ 28147-89 сообщение m и $c = K(m)$ и передает информацию абоненту.

Дальнейшая пересылка может идти в обоих направлениях, так как у абонента и центра имеется общий ключ. Так, получив шифртекст c, абонент применяет $K(c) = m$.

Таким образом, генерация и использование ключей осуществляются только на период одного сеанса связи. Особые действия при компрометации ключей не требуются, поскольку потеря одного аппарата может быть легко блокирована посредством определения его адреса, обязательного к использованию при обмене информацией между абонентами.

Криптопроцессор не имеет встроенного датчика случайных (псевдослучайных) чисел. Сеанс связи "точка — точка" с использованием главного ключа.

Для повышения надежности криптографической схемы, а также уменьшения нагрузки на криптографический процессор пакетного контроллера в некоторых случаях целесообразно исключить датчик случайных (псевдослучайных) чисел из криптографического процессора пакетного контроллера абонента. В этом случае только в центре имеется ДСЧ (ДПСЧ), с помощью которого до начала информационного обмена на ключевой носитель генерируется главный ключ (ГК) $K_{\text{ГК}}$. Один экземпляр ГК передается абоненту системы A, другой остается в центре. Для повышения стойкости ГК используется исключительно для передачи сеансового ключа.

На первом этапе центр генерирует сеансовый ключ $K_{\text{с}}$, шифрует ключ $s = K_{\text{ГК}}(K_{\text{с}})$, затем посредством имитовставки получает сообщение s'. Абонент, получив от центра сообщение s', убирает имитовставку, затем расшифровывает сообщение, являющееся сеансовым ключом $K_{\text{с}} = K_{\text{ГК}}(s)$. Используя сеансовый ключ $K_{\text{с}}$, абонент шифрует сообщение m и получает соответствующее ему зашифрованное сообщение $c = K_{\text{с}}(m)$ и передает в центр. Центр, получив шифрованное сообщение c, применяет $m = K_{\text{с}}(c)$, тем самым получая исходное сообщение m.

В случае компрометации ключа требуется смена главного ключа, ключи считаются скомпрометированными и использованию более не подлежат. Ключевой носитель может иметь несколько ГК, различные главные ключи могут располагаться на нескольких ключевых носителях и использоваться в различных режимах работы криптопроцессора.

Криптопроцессор не имеет встроенного датчика случайных (псевдослучайных) чисел. Сеанс связи "точка — точка" с использованием одноразового блокнота.

С другой стороны, можно несколько упростить предыдущую схему, уменьшив количество генераций случайных (псевдослучайных) последовательностей путем использования схемы "одноразовый блокнот". Так, в центре имеется ДСЧ (ДПСЧ), с помощью которого до начала информационного обмена на ключевой носитель генерируется гамма шифра (Г). Один экземпляр гаммы шифра Г передается абоненту системы, другой остается в центре в виде блокнота U.

Таким образом, последовательность установки общего ключа будет следующей: центр генерирует сеансовый ключ $K_{\text{с}}$ и режим гаммирования R, затем применяет гаммирование (например, функция "исключающего или", XOR) по режиму R, $s = K_{\text{с}} \oplus \Gamma(R)$ посредством имитовставки получает сообщение s'. Затем центр передает $\langle s', R\rangle$ абоненту системы.

Абонент, получив пару сообщений $\langle s', R\rangle$, убирает имитовставку, затем применяет функцию, обратную гаммированию $K_{\text{с}} = s \oplus \Gamma(R)$, сообщение, являющееся сеансовым ключом. Используя сеансовый ключ $K_{\text{с}}$, абонент шифрует сообщение m и $c = K_{\text{с}}(m)$ и передает в центр. На последнем этапе центр получает шифртекст c и применяет $m = K_{\text{с}}(c)$.

В случае компрометации ключевого носителя, хранящего гамму, требуется смена гаммы, ключи считаются скомпрометированными и использованию более не подлежат. Ключевой носитель может иметь любое количество сколько угодно больших гамм (единственным ограничением является объем имеющейся памяти) и множество режимов гаммирования, что существенно повышает стойкость.

Очевидно, что при учете значительных ограничений на ресурсы криптографического процессора необходимо оценить вычислительные затраты и требуемые объемы

памяти для реализации криптографических преобразований. Подробный расчет приведен в [5].

Заключение

В ходе проведения работы были получены результаты по реализации криптографических протоколов в системах с ограниченными параметрами.

Список литературы

[1] Мещеряков Р.В., Шелупанов А.А., Росошек С.К., Бондарчук С.С. Встраивание криптографических функций в систему связи с ограниченными ресурсами // Вопр. защиты информации. М.: ФГУП ВИНИ. 2004. № 2. С. 22–25.

[2] ГОСТ 28147-89. Системы обработки информации. Защита криптографическая. Алгоритм криптографического преобразования.

[3] Аграновский А.В., Балакин А.В, Хади Р.А. Классические шифры и методы их криптоанализа // Информ. технологии. 2001. № 10.

[4] Шнайдер Б. Прикладная криптография. М.: Триумф, 2002.

[5] Мещеряков Р.В., Шелупанов А.А., Росошек С.К., Сонькин М.А. Защищенная сеть передачи данных на основе пакетного контроллера // Вест. Сиб. гос. аэрокосм. ун-та им. М.Ф. Решетнева. 2006. № 10. С. 171–175.

Поступила в редакцию 23 августа 2007 г.