УДК 621.391.7:336.71(075.8)

И.В. ОЛЕШКО, И.Д. ГОРБЕНКО, д-р техн. наук

СРАВНИТЕЛЬНЫЙ АНАЛИЗ ПРОТОКОЛОВ СТРОГОЙ АУТЕНТИФИКАЦИИ

Введение

Аутентификация сообщений представляет собой процедуру, обеспечивающую связывающимся сторонам возможность проверки аутентичности (подлинности) получаемых сообщений. Двумя важными задачами аутентификации является проверка того, что содержимое сообщения не было изменено, и того, что сообщение прибыло именно из того источника, о котором информирует сообщение [1 – 3]. Одной из распространенных схем аутентификации является простая аутентификация, которая основана на применении традиционных многоразовых паролей с одновременным согласованием средств их использования и обработки. Идея строгой аутентификации, реализуемая в криптографических протоколах, заключается в следующем. Проверяемая (доказывающая) сторона доказывает свою подлинность проверяющей стороне, демонстрируя знание какого-либо секрета, который, например, может быть предварительно распределен безопасным способом между сторонами аутентификационного обмена. Доказательство знания секрета осуществляется с помощью последовательности запросов и ответов с использованием криптографических методов и средств.

Существенным является тот факт, что доказывающая сторона демонстрирует только знание секрета, но сам секрет в ходе аутентификационного обмена не раскрывается. Это обеспечивается посредством ответов доказывающей стороны на различные запросы проверяющей стороны. При этом результирующий запрос зависит только от пользовательского секрета и начального запроса, который обычно представляет собой произвольно выбранное в начале протокола большое число.

В большинстве случаев строгая аутентификация заключается в том, что каждый пользователь аутентифицируется по признаку владения своим секретным ключом. Иначе говоря, пользователь имеет возможность определить, владеет ли его партнер по связи надлежащим секретным ключом и может ли он использовать этот ключ для подтверждения того, что он действительно является подлинным партнером по информационному обмену.

В соответствии с рекомендациями стандарта X.509 различают процедуры строгой аутентификации следующих типов [4]:

- односторонняя аутентификация;
- двусторонняя аутентификация;
- трехсторонняя аутентификация.

Односторонняя аутентификация предусматривает обмен информацией только в одном направлении. Данный тип аутентификации позволяет:

1. Подтвердить подлинность только одной стороны информационного обмена;
2. Обнаружить нарушение целостности передаваемой информации;
3. Обнаружить проведение атаки типа «повтор передачи»;
4. Гарантировать, что передаваемыми аутентификационными данными может воспользоваться только проверяющая сторона.

Двусторонняя аутентификация по сравнению с односторонней содержит дополнительный ответ проверяющей стороны доказывающей стороне, который должен убедить ее, что связь устанавливается именно с тем партнером, которому были предназначены аутентификационные данные. Трехсторонняя аутентификация содержит дополнительную передачу данных от доказывающей стороны проверяющей. Такой подход позволяет отказаться от использования меток времени при проведении процедуры аутентификации. В зависимости от используемых криптографических алгоритмов протоколы строгой аутентификации можно разделить на следующие группы [2, 3]:

- на симметричных алгоритмах шифрования;

• на алгоритмах электронной цифровой подписи;

• на использовании криптографического контрольного значения;

• основанные на нулевых знаниях;

• на сертификатах с использованием преобразований в группах точек эллиптической кривой.

Введем следующие обозначения:

TokenAB – маркер, который отсылается предъявителем А проверяющему В;

eK(Z) – результат зашифрования данных Z по алгоритму симметричного шифрования, используя ключ К;

T_X – метка времени, созданная объектом X;

N_X – порядковый номер, созданный объектом X;

A – идентификатор объекта *A;*

B – идентификатор объекта *B;*

$sS_x(Z)$ – электронная цифровая подпись, полученная в результате преобразования над данными Z, используя личный ключ подписи S_x;

R_x – случайное число, созданное объектом x.

Цель статьи – сравнительный анализ основних протоколов строгой аутентификации на основании условных и безусловных критериев.

Строгая аутентификация, основанная на симметричных алгоритмах

Для работы протоколов аутентификации, построенных на основе симметричных алгоритмов, необходимо, чтобы проверяющий и доказывающий с самого начала имели один и тот же секретный ключ. Для закрытых систем с небольшим количеством пользователей каждая пара пользователей может заранее разделить его между собой. В больших распределенных системах, применяющих технологию симметричного шифрования, часто используются протоколы аутентификации с участием доверенного сервера, с которым каждая сторона разделяет знание ключа. Такой сервер распределяет сеансовые ключи для каждой пары пользователей всякий раз, когда один из них запрашивает аутентификацию другого.

Требования, предъявляемые к механизмам аутентификации, приведены в стандарте [5]. Все эти требования должны быть выполнены, в противном случае появляется возможность угрозы компрометации процесса аутентификации или даже невозможность выполнения аутентификации.

Ниже приводятся два примера протоколов аутентификации, специфицированных в ISO/IEC 9798-2 [5]. Эти протоколы предполагают предварительное распределение разделяемых секретных ключей.

Рассмотрим следующие варианты аутентификации:

1. Односторонняя аутентификация с одним проходом;

2. Двусторонняя аутентификация с тремя проходами.

Односторонняя аутентификация с одним проходом. В этом протоколе аутентификации предъявитель А инициирует процесс, а объект В проверяет его аутентичность. Уникальность и своевременность обеспечивается путем генерации и проверки метки времени и порядкового номера.

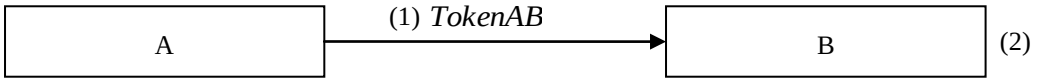

Рис. 1. Протокол односторонней аутентификации с одним проходом

Маркер (*TokenAB*), который отсылается предъявителем А проверяющему В, имеет такую структуру:

$$TokenAB = Text2 \,||\, eK_{AB}\begin{pmatrix} T_A \\ N_A \end{pmatrix} ||\, B \,||\, Text1\Big). \quad (1)$$

В маркере предъявитель А в качестве изменяемого во времени параметра использует или порядковый номер N_A, или метку времени T_A. Присутствие идентификатора В в маркере *TokenAB* не обязательно, если отсутствует возможность осуществления атаки типа маскарад.

Протокол осуществляется следующим образом:

1) Объект А генерирует и отправляет *TokenAB* объекту В.

2) После получения сообщения, которое содержит *TokenAB*, объект В проверяет *TokenAB* путем расшифрования зашифрованной части. Затем проверяется корректность идентификатора В, если он есть, а также метки времени или порядкового номера.

Двусторонняя аутентификация с тремя проходами. В этом протоколе аутентификации уникальность и / или своевременность обеспечиваются путем генерации и проверки случайного числа R_B. Протокол представлен на рис. 2.

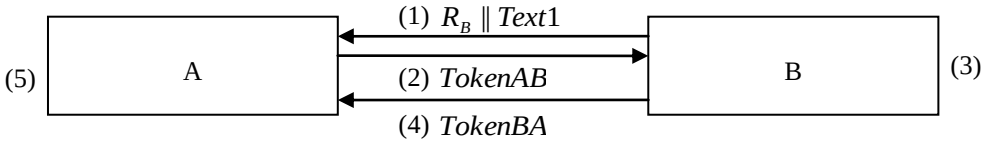

Рис. 2. Протокол двусторонней аутентификации с тремя проходами

Маркеры имеют такую структуру:

$$TokenAB = Text3 \,||\, eK_{AB}\left(R_A \,||\, R_B \,||\, B \,||\, Text2\right) \quad (2)$$

$$TokenBA = Text5 \,||\, eK_{AB}\left(R_B \,||\, R_A \,||\, Text4\right). \quad (3)$$

Протокол осуществляется следующим образом:

1) Объект В генерирует случайное число R_B и отправляет его объекту А.

2) Объект А генерирует случайное число R_A, формирует и отправляет объекту В маркер *TokenAB*

3) Объект В, получив сообщение, содержащее *TokenAB*, проверяет этот маркер путем расшифрования его зашифрованной части. Затем проверяет:

- корректность идентификатора В, если он есть.
- соответствие случайного числа R_B и случайного числа, которое содержится в маркере *TokenAB*.

4) Объект В генерирует и отправляет маркер *TokenBA* объекту А

5) Объект А, получив сообщение, содержащее *TokenBA*, проверяет этот маркер путем расшифрования его зашифрованной части. Затем проверяет: корректность идентификатора А, если он есть, случайного числа R_B и случайного числа, которое содержится в маркере *TokenBA*, соответствие случайного числа R_A, которое было отослано объекту В на шаге 2 и случайного числа, которое содержится в маркере *TokenBA*.

Протоколы строгой аутентификации, основанные на использовании электронной цифровой подписи

Рассмотрим следующие варианты протоколов аутентификации [6]:

1. односторонняя аутентификация с одним проходом;
2. двусторонняя аутентификация с тремя проходами.

Односторонняя аутентификация с одним проходом. Механизм изображен на рис. 3.

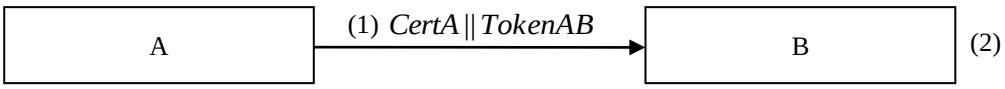

Рис. 3. Протокол односторонней аутентификации с одним проходом

Маркер (*TokenAB*), который отсылается предъявителем А проверяющему В, имеет такую структуру:

$$TokenAB = \begin{matrix} T_A \\ N_A \end{matrix} || B || Text2 || sS_A\left(\begin{matrix} T_A \\ N_A \end{matrix} || B || Text1\right) \quad (4)$$

Протокол осуществляется следующим образом:

1) Объект А генерирует и отправляет *TokenAB* объекту В, а также свой сертификат (не обязательно).

2) После получения сообщения, которое содержит *TokenAB*, объект В выполняет следующее:

- убеждается в том, что обладает действующим открытым ключом объекта А в виде сертификата или любых других средств;
- проверяет *TokenAB* путем:
- проверяет электронную цифровую подпись, которая содержится в маркере;
- проверяет корректность метки времени или порядкового номера;
- сравнивает идентичность значения поля идентификатора В в подписанных данных маркера *TokenAB* и идентификатора объекта В.

Двусторонняя аутентификация с тремя проходами. В этом протоколе аутентификации уникальность и/или своевременность обеспечивается путем генерации и проверки случайных чисел. Механизм изображен на рис.4.

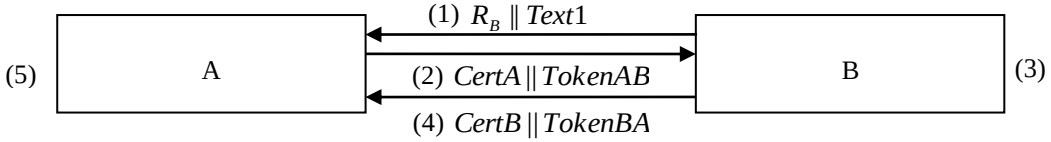

Рис. 4. Протокол двусторонней аутентификация с тремя проходами

Маркеры имеют следующую структуру:

$$TokenAB = R_A \| R_B \| B \| Text3 \| sS_A(R_A \| R_B \| B \| Text2) \quad (5)$$

$$TokenBA = R_B \| R_A \| A \| Text5 \| sS_B(R_B \| R_A \| A \| Text4) \quad (6)$$

Присутствие идентификатора В в маркере *TokenAB* и присутствие идентификатора А в маркере *TokenBA* необязательно и зависит от окружения, в котором осуществляется протокол аутентификации.

Протокол осуществляется следующим образом:

1) Объект В генерирует случайное число R_B и отправляет его объекту А (также может быть отправлено значение текстового поля *Text*1).

2) Объект А формирует и отправляет объекту В маркер *TokenAB*, а также свой сертификат (не обязательно).

3) После получения сообщения, которое содержит *TokenAB*, объект В выполняет следующие действия:

- убеждается в том, что обладает действующим открытым ключом объекта А в виде сертификата или любых других средств;
- проверяет *TokenAB* путем:

– проверяет электронную цифровую подпись, которая содержится в маркере;

– проверяет соответствие случайного числа R_B, которое было отослано объекту А на шаге (1) и случайного числа, которое содержится в подписанных данных маркера *TokenAB*;

– сравнивает идентичность значения поля идентификатора В в подписанных данных маркера *TokenAB* и идентификатора объекта В.

4) Объект В генерирует и отправляет объекту А маркер *TokenBA*, а также свой сертификат (необязательно).

5) После получения сообщения, которое содержит *TokenBA*, объект А выполняет действия, аналогичные тем, которые выполнял объект В на 3 шаге. Дополнительно происходит проверка соответствия случайного числа R_B, которое содержится в маркере *TokenBA* и случайного числа, полученного на шаге 1.

Протоколы строгой аутентификации, основанные на использовании криптографического контрольного значения

Рассматриваются следующие варианты протоколов [7]:

- односторонняя аутентификация с одним проходом;
- двусторонняя аутентификация с тремя проходами.

Односторонняя аутентификация с одним проходом. Механизм изображен на рис.1.

Маркер (*TokenAB*), который отсылается предъявителем А проверяющему В, имеет такую структуру:

$$TokenAB = \begin{matrix} T_A \\ N_A \end{matrix} \| Text2 \| f_{K_{AB}}\left(\begin{matrix} T_A \\ N_A \end{matrix} \| B \| Text1\right) \quad (7)$$

В данном протоколе в качестве изменяемого во времени параметра используется или порядковый номер N_A, или метка времени $T_{A.}$. Как определяется в ISO/ IEC 9798-1, $f_k(X)$ – криптографическое контрольное значение, где К – секретный ключ, Х – случайная строка данных. Протокол осуществляется следующим образом:

1. Объект А генерирует и отправляет *TokenAB* объекту В.

2. После получения сообщения, которое содержит *TokenAB*, объект В проверяет *TokenAB* путем проверки метки времени или порядкового номера, вычисляя $f_{K_{AB}}\left(\begin{matrix} T_A \\ N_A \end{matrix} \| B \| Text1\right)$ и сравнивая полученное значение с криптографическим контрольным значением, которое содержится в маркере. Вместе с этим проверяется корректность идентификатора В (если он есть).

Двусторонняя аутентификация с тремя проходами. Механизм изображен на рис.2. Маркеры имеют следующую структуру:

$$TokenAB = R_A \| Text3 \| f_{K_{AB}}\left(R_A \| R_B \| B \| Text2\right) \quad (8)$$

$$TokenBA = Text5 \| f_{K_{AB}}\left(R_B \| R_A \| Text4\right) \quad (9)$$

Протокол осуществляется следующим образом:

1) Объект В генерирует случайное число R_B и отправляет его объекту А (также может быть отослано текстовое поле *Text1*).

2) Объект А генерирует случайное число R_A, формирует и отправляет объекту В маркер *TokenAB*

3) Объект В, получив сообщение, содержащее *TokenAB*, проверяет этот маркер, вычисляя значение $f_{K_{AB}}\left(R_A \| R_B \| B \| Text2\right)$ и сравнивая полученное значение с криптографическим контрольным значением, которое содержится в маркере. Таким образом, проверяется корректность идентификатора В (если он есть), а также соответствие случайного числа R_B, которое было отослано объекту А на шаге (1) и случайного числа, которое было использовано для генерации маркера *TokenAB*.

4) Объект В генерирует и отправляет маркер *TokenBA* объекту А.

5) Объект А, получив сообщение, которое содержит маркер *TokenBA*, проверяет его, вычисляя значение $f_{K_{AB}}(R_B \parallel R_A \parallel Text4)$ и сравнивая это значение с криптографическим контрольным значением, которое содержится в маркере. Таким образом, проверяется, что случайное число R_B, которое было получено от объекта В на шаге (1), и случайное число R_A, отосланное объекту В на шаге (2), были использованы для генерации маркера *TokenBA*.

Протоколы строгой аутентификации, которые используют методы, основанные на нулевых знаниях

Требования, выдвигаемые к данной группе протоколов, а также обоснование выбора основных параметров приведены в стандарте [8].

Рассмотрим следующие варианты протоколов:

- механизмы, основанные на использовании данных идентификации;
- механизмы, основанные на сертификатах с использованием системы асимметричного шифрования;
- механизмы, основанные на сертификатах с использованием дискретных логарифмов.

Механизмы, основанные на использовании данных идентификации. При выполнении однонаправленного механизма аутентификации предъявитель А инициирует процесс, а объект В проверяет его аутентичность. Для корректного выполнения механизма важно обеспечить объекта В данными идентификации объекта А, которые добавляются к одному из информационных обменов этого протокола. Одна итерация процедуры аутентификации приведена на рис.5.

Форма первого маркера ($TokenAB_1$) выглядит следующим образом:

$$TokenAB_1 = h(W \parallel Text) \tag{10}$$

где W – это доказательство, h – функция хеширования, а $Text$ – необязательное текстовое поле. Форма второго маркера имеет следующий вид:

$$TokenAB_2 = D \tag{11}$$

где D – ответ.

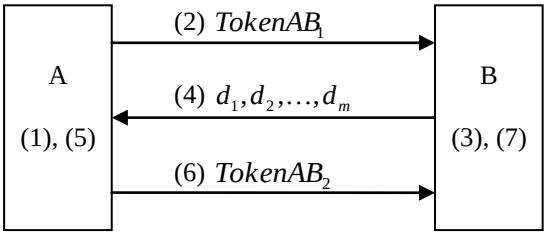

Рис. 5. Механизмы, основанные на использовании данных идентификации

Для каждого применения данного механизма подобная процедура аутентификации должна выполняться t раз. Проверяющий В должен считать, что предъявитель А прошел аутентификацию только при условии, что все t итераций завершились успешно. Протокол выполняется следующим образом:

1) Объект А, который владеет личной информацией аккредитации C_{A1}, C_{A2},..., C_{Am}, выбирает случайное число r, которое должно быть целым и удовлетворять неравенству $1 \le r \le n-1$. Это число сохраняется в тайне объектом А. Объект А вычисляет доказательство W:

$$W = r^v \ mod^* \ n \tag{12}$$

2) Объект А отправляет $TokenAB_1$ объекту В.

3) Получив маркер $TokenAB_1$, объект В должен случайным образом выбрать последовательность целых чисел $d_1,d_2,\ldots d_m$, где $0<d_i< \nu-1$.

4) Объект В отправляет запрос $d_1,d_2,\ldots d_m$ объекту А.

5) Получив запрос $d_1,d_2,\ldots d_m$, объект А должен посчитать ответ D из секретного значения r и личной информации аккредитации $C_{A1},C_{A2},\ldots,C_{Am}$ по следующей формуле:

$$D = r\prod_{i=1}^{m}(C_{Ai})^{d_i} \bmod^* n \tag{13}$$

6) Объект А отправляет маркер $TokenAB_2$ объекту В.

7) Получив ответ D, объект В должен выполнить следующие вычисления:

- проверяет, что $0<D<\frac{n}{2}$. Если это не так, объект В не аутентифицирует объект А.
- вычисляет избыточные данные идентификации $J_{A1}, J_{A2}, \ldots, J_{Am}$, из данных идентификации $I_{A1}, I_{A2},\ldots I_{Am}$
- вычисляет значение W’ по формуле:

$$W' = D^{\nu}\prod_{i=1}^{m}(J_{Ai})^{d_i} \; mod^* \; n \tag{14}$$

если W было отослано при первом обмене процедуры, то объект В проверяет, что вычисленное значение W’ = W. Если h(W||Text) было отослано при первом обмене протокола, то объект В проверяет, что h(W’||Text) = h(W||Text). Если проверка прошла успешно, то вся итерация завершена успешно. В других случаях В не аутентифицирует объект А.

Механизмы, основанные на сертификатах с использованием системы ассиметричного шифрования. Ниже приведены обмены, которые необходимо совершать при выполнении однонаправленного механизма аутентификации между предъявителем А и объектом В, который проверяет его аутентичность. Механизм аутентификации приведен на рис.6.

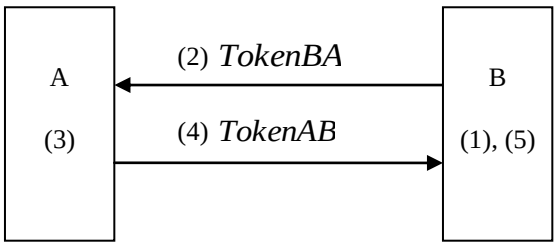

Рис. 6. Механизм, основанный на сертификатах
с использованием системы ассиметричного шифрования

Форма первого маркера (*TokenBA*), который отправляется проверяющим предъявителю:

$$TokenBA = d \tag{15}$$

где d – запрос.

Форма маркера (*TokenAB*), который отправляется предъявителем проверяющему выглядит следующим образом:

$$TokenAB = D \tag{16}$$

где D – это ответ.

Протокол выполняется следующим образом:

1) Объект B выбирает случайное число r. Это число сохраняется в тайне объектом B. Затем объект B вычисляет h(r). Случайное число r должно быть выбрано таким способом, чтобы r||h(r) принадлежало к области P_A, открытого преобразования зашифрования объекта A. Объект B вычисляет запрос d по формуле

$$d = P_A(r \parallel h(r)) \tag{17}$$

2) Объект B отправляет *TokenBA* объекту A.

3) Получив маркер *TokenBA*, объект A должен выполнить следующее:

- объект A получает значение *r* путем вычисления

$$r \,||\, h(r) = S_A(d) \tag{18}$$

где S_A – криптографическое преобразование расшифрования объекта A;

- объект A вычисляет значение $h(r)$ из полученного на шаге 3 значения r и сравнивает его со значением, полученным из маркера *TokenBA*. Если они не равны, A прекращает выполнение протокола, в другом случае A устанавливает значение ответа *D*: $D = r$.

4) Объект A отправляет $TokenAB = D$ объекту B.

5) Получив маркер *TokenAB*, объект B сравнивает значение ответа *D* и *r*. Если $r \neq D$, то считается, что механизм был провален, объект A не прошел аутентификацию. Если же $r = D$, то объект B признает аутентичность объекта A.

Механизмы, основанные на сертификатах с использованием дискретных логарифмов. Механизм аутентификации приведен на рис.7.

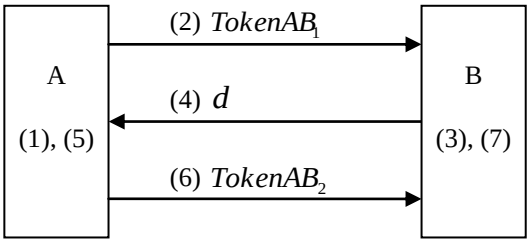

Рис. 7. Механизм, основанный на сертификатах с использованием дискретных логарифмов

Протокол выполняется следующим образом:

1) Объект A выбирает случайное число $1 \le r \le q$. Это число сохраняется в тайне объектом A. Объект A вычисляет доказательство W

$$W = g^r \bmod p \tag{19}$$

2) Объект A посылает $TokenAB_1$ объекту B. Маркер $TokenAB_1$ должен быть равен или W или h(W||Text);

3) Получив маркер $TokenAB_1$, объект B случайным образом выбирает $0 \le d \le q$;

4) Объект B отправляет запрос d объекту A;

5) Получив запрос d, объект A должен вычислить ответ D из секретного значения r и личного ключа z_A объекта A по формуле:

$$D = r - dz_A \bmod q \tag{20}$$

6) Объект A отправляет маркер $TokenAB_2$ объекту B.

7) Получив ответ D, объект B должен выполнить следующее:

- проверяет, что $0 < D < q$. И если это не так, то B отбраковывает A.
- объект B вычисляет W' по формуле

$$W' = (y_A)^d \, g^D \bmod p \tag{21}$$

Если W было отослано при первом обмене, то объект B проверяет, что значение W'=W. Если $h(W \parallel Text)$ было отослано при первом обмене, то B проверяет, значение $h(W' \parallel Text) = h(W \parallel Text)$.

Протоколы строгой аутентификации, основанные на сертификатах с использованием преобразований в группах точек эллиптической кривой

Рассмотрим согласование ключей типа Диффи-Гелмана (KANIDH) [9]. Этот механизм обеспечивает автономное установление разделяемого секрета между двумя объектами A и B.

Механизм протокола заключается в следующем:

1) Формирование ключа (A): объект A вычисляет разделяемый ключ, используя свой личный ключ для разделения ключей d_A и открытый ключ для разделения ключей P_A объекта A таким образом:

$$K_{AB} = (d_B \cdot l)(hP_A) \tag{22}$$

Свойства протокола:

- число проходов 0;
- механизм обеспечивает взаимную неявную аутентификацию ключа.

Критерии оценки криптографических протоколов

Криптографические протоколы будем оценивать на основании условных и безусловных критериев.

Безусловные критерии:

1. Надежность математической базы.
2. Практическая защищенность криптопреобразований от силовых и аналитических атак.
3. Реальная защищенность от всех известных и потенциально возможных криптоаналитических атак.
4. Статистическая безопасность криптографических преобразований.
5. Отсутствие слабых личных ключей.

Все обозначенные критерии не могут выполняться частично. Т.е. протокол может или удовлетворять поставленным условиям, или нет. При этом невыполнение хотя бы одного условия должно приводить к отказу от использования такого криптопреобразования, т.к. криптопреобразование является нестойким и его использование может привести к полному взлому системы криптографической защиты информации.

Условные критерии, используемые для сравнения криптографических протоколов:

- наличие и вид ключевой аутентификации. Обозначим этот критерий K_{y1};
- наличие и вид аутентификации субъекта. Обозначим этот критерий K_{y2};
- новизна ключей. Обозначим этот критерий K_{y3};
- управление ключами. Обозначим этот критерий K_{y4};
- эффективность протокола. Обозначим этот критерий K_{y5};
- криптоживучесть ключей. Обозначим этот критерий K_{y6};
- скорость выполнения элементарных операций и протокола в целом. Обозначим этот критерий K_{y7};

• уровень защищенности при реализации разных атак при разных условиях и отклонении свойств общесистемных параметров. Обозначим этот критерий K_{y8}.

Сравнительный анализ протоколов аутентификации

Сравнительный анализ протоколов аутентификации выполнен с использованием условных и безусловных критериев. Безусловные критерии выполняются для всех рассматриваемых протоколов. Анализ условных критериев приведен в таблице.

Таким образом, из таблицы можно сделать вывод о том, что более предпочтительным протоколом аутентификации является протокол, основанный на сертификатах с использованием преобразований в группе точек ЭК. За ним можно поставить протокол, основанный на использовании данных идентификации, затем протокол, основанный на сертификатах с использованием системы асимметричного шифрования и протокол, основанный на сертификатах с использованием дискретных логарифмов.

Протоколы / Критерии	Протокол, основанный на использовании данных идентификации	Протокол, основанный на сертификатах с использованием дискретных логарифмов	Протокол, основанный на сертификатах с использованием системы асимметричного шифрования	Протокол, основанный на сертификатах с использованием преобразований в группе точек ЭК
Аутентификация субъекта	Субъект А для субъекта В	Субъект А для субъекта В	Субъект А для субъекта В	Субъект А для субъекта В
Аутентификация ключа	Явная от А к У	Явная от А к У	Явная от А к У	Явная от А к У
Вид аутентификации субъекта	Односторонняя абонента А к У	Односторонняя абонента А к У	Односторонняя абонента А к У	Односторонняя абонента А к У
Вид аутентификации ключа	Односторонняя аутентификация ключа абонента А	Односторонняя аутентификация ключа абонента А	Односторонняя аутентификация ключа абонента А	Односторонняя аутентификация ключа абонента А
Наличие подтверждения ключа	Подтверждение ключа абонента А, C_A	Подтверждение ключа абонента А, Z_A	Подтверждение ключа абонента А, S_A	Подтверждение ключа абонента А, d_A
Новизна ключей	Абонент А владеет секретной информацией аккредитации C_A, а В знает только открытый ключ J_A C_A-статичный J_A- может меняться	Абонент А владеет секретным ключом Z_A, а В знает только открытый ключ Y_A, потому нет новизны ключей	Абонент А имеет собственное криптографическое преобразование S_A, а В знает только область открытого преобразования P_A, потому нет новизны ключей. P_A – определено однозначно, а S_A может меняться	Абонент А имеет секретный ключ d_A, а В знает только открытый ключ Y_A, потому нет новизны ключей, т.к. невозможно было бы вычислить значение W'
Управление ключевыми данными	Значение информации аккредитации C_A определяет доверенная сторона	По усмотрению сторон протокола	По усмотрению сторон протокола	По усмотрению сторон протокола

Защита от атаки типа «повтор ранее переданного сообщения»	Происходит за счет случайного числа r и последовательности целых чисел $d_1, d_2, d_3 K d_m$	Происходит за счет случайного числа r и случайного целого числа d	Происходит за счет случайного числа r	Происходит за счет случайного числа r и случайного целого числа d
Число обменов сообщениями	3	3	2	3
Сложность вычислений	1 операция секретного преобразования, 1 операция вычисления остаточных данных идентификации J_A и 1 операция открытого преобразования (3)	1 операция секретного преобразования, 1 операция открытого преобразования (2)	1 операция секретного преобразования, 1 операция открытого преобразования (2)	1 операция секретного преобразования 1 операция открытого преобразования (2)
Возможность использования предыдущих вычислений	Нет	Нет	Нет	Нет
Требования к 3-й стороне	Генерация для А информации аккредитации	Участники протокола сами решают, кто изготовляет пару ключей и если это третья сторона, то соответственно это генерация открытого и секретного ключа.	Участники протокола сами решают, кто изготовляет пару ключей и если это третья сторона, то соответственно это генерация открытого и секретного ключа.	Участники протокола сами решают, кто изготовляет пару ключей и если это третья сторона, то соответственно это генерация открытого и секретного ключа.
Криптостойкость ключа	Обеспечивается криптостойкость J_A ,при отсутствии компрометации секретной информации аккредитации C_A	Обеспечивается криптостойкость Y_A, при отсутствии компрометации секретной информации аккредитации Z_A	Обеспечивается криптостойкость P_A, при отсутствии компрометации секретной информации аккредитации S_A	Обеспечивается криптостойкость Y_A, при отсутствии компрометации секретной информации аккредитации d_A
Сложность реализации атак «полное раскрытие»	Субэкспоненциальная	Субэкспоненциальная	Субэкспоненциальная	Экспоненциальная
Неопровержимость	Неопровержимость объекта А происходит за счет C_A	Неопровержимость объекта А происходит за счет Z_A	Неопровержимость объекта А происходит за счет S_A	Неопровержимость объекта А происходит за счет d_A

Выводы

Расширение круга пользователей распределенных систем и усложнение задач, решаемых с помощью подобных сетей, привели к осознанию первостепенной важности проблем, связанных с обеспечением информационной безопасности при межсетевом взаимодействии. Поэтому одной из важных задач обеспечения безопасности информации является разработка

и анализ протоколов аутентификации, позволяющих одной стороне (проверяющему) убедиться в идентичности другой стороны (доказывающего). Одной из распространенных схем аутентификации является простая аутентификация, которая основана на применении традиционных многоразовых паролей с одновременным согласованием средств их использования и обработки. Одна из уязвимостей протоколов простой аутентификации заключается в том, что после того, как доказывающий передаст проверяющему свой пароль, проверяющий может выдать себя за проверяемого. Идея строгой аутентификации заключается в том, что проверяемая (доказывающая) сторона доказывает свою подлинность проверяющей стороне, демонстрируя знание какого-либо секрета, не раскрывая его.

В настоящее время на фоне огромного количества криптографических протоколов становится вопрос выбора того или иного протокола строгой аутентификации, который больше всего удовлетворяет поставленным требованиям. Выбрать такой протокол возможно только в случае полного анализа и сравнения характеристик криптографических протоколов.

В статье предлагается сравнивать криптографические протоколы аутентификации на основании условных и безусловных критериев. Основная задача безусловных критериев – выполнить анализ на предмет возможности применения выбранного криптографического протокола в принципе. Задача условных критериев – вывести числовые оценки для каждого криптографического протокола.

На основании анализа протоколов строгой аутентификации, был сделан вывод о том, что наилучшим протоколом аутентификации является протокол, основанный на сертификатах с использованием преобразований в группе точек ЭК.

Список литературы: 1. *Иванов, М. А*. Информации в компьютерных системах и сетях. М.: КУДИЦ-ОБРАЗ, 2001. – 368 с. 2. *Соколов, А.В*. Защита информации в распределенных корпоративных сетях и системах / А.В. Соколов, В.Ф. Шаньгин. – М. : ДМК Пресс, 2002. – 656 с. 3. *Шнайер, Б*. Прикладная криптография. Протоколы, алгоритмы, исходные тексты на языке Си / Б. Шнайер. – М. : ТРИУМФ, 2003. – 816 с. 4. *X509-2001 – 4*. Використання каталогом основних положень сертифікації відкритого ключа та сертифікації атрибутів. 5. *ДСТУ* ISO/IEC 9798-2. Методи захисту. Автентифікація об’єктів. Частина 2: Механізми, що ґрунтуються на використанні алгоритмів симетричного шифрування. 6. *ДСТУ* ISO/IEC 9798-3. Методи захисту. Автентифікація об’єктів. Частина 3: Протоколи, що ґрунтуються на використанні електронного цифрового підпису. 7. *ДСТУ* ISO/IEC 9798-4. Методи захисту. Автентифікація об’єктів. Частина 4: Протоколи, що ґрунтуються на використанні функцій обчислення криптографічного контрольного значення. 8. *ISO/IEC* 9798-5. Методи захисту. Автентифікація об’єктів. Частина 5: Протоколи, що використовують методи які грунтуються на нульових знаннях. 9. *ISO/IEC* 15946-3:2002, IDT. Методи захисту криптографічні перетворення, що ґрунтуються на еліптичних кривих .

Харьковский национальный
университет радиоэлектроники

Поступила в редколлегию 20.09.2012