Методы обеспечения целостности информации
Протоколы электронной цифровой подписи.

ФИО преподавателя: Ермакова А.Ю.

e-mail: ermakova_a@mirea.ru

Протоколы электронной подписи.

• Протоколы ЭЦП с одной стороны относят к протоколам аутентификации, т.к. гарантируют, что сообщение поступило от достоверного отправителя, а с другой стороны к протоколам контроля целостности, т.к. гарантируют, что сообщение пришло в неискаженном виде. Более того, получатель в дальнейшем может использовать ЭЦП как доказательство достоверности сообщения третьим лицам (арбитру) в том случае, если отправитель впоследствии попытается отказаться от него.

Протоколы электронной подписи.

- Говоря о схеме цифровой подписи, обычно имеют в виду следующую **классическую ситуацию**:
  - - отправитель знает содержание сообщения, которое он подписывает;
  - - получатель, зная открытый ключ проверки подписи, может проверить правильность подписи полученного сообщения в любое время без какого-либо разрешения и участия отправителя;
  - - безопасность схемы подписи гарантируется.

Протоколы электронной подписи.
Электронная подпись на основе алгоритма RSA

• Пусть, как и раньше, пользователь А хочет передать пользователю Б сообщение, состоящее из нескольких блоков m_i. Перед началом сеанса связи абоненты генерируют открытые и закрытые ключи. В результате каждый пользователь имеет свои собственные открытый (состоящий из двух частей) и закрытый ключи. Затем пользователи обмениваются открытыми ключами. Это подготовительный этап протокола.

Протоколы электронной подписи. Электронная подпись на основе алгоритма RSA

Основная часть протокола состоит из следующих шагов.

1. Сначала пользователь А вычисляет числа $c_i = m_i^{e_A} \, mod \, N_A$, то есть шифрует сообщение своим закрытым ключом. В результате этих действий пользователь А подписывает сообщение.
2. Затем пользователь А вычисляет числа $g_i = c_i^{d_Б} \, mod \, N_Б$, то есть шифрует то, что получилось на шаге 1 открытым ключом пользователя Б. На этом этапе сообщение шифруется, чтобы никто посторонний не мог его прочитать.
3. Последовательность чисел g_i передается к пользователю Б.
4. Пользователь Б получает g_i и вначале вычисляет последовательно числа $c_i = g_i^{e_Б} \, mod \, N_Б$, используя свой закрытый ключ. При этом сообщение расшифровывается.
5. Затем Б определяет числа $m_i = c_i^{d_A} \, mod \, N_A$, используя открытый ключ пользователя А. За счет выполнения этого этапа производится проверка подписи пользователя А.

Протоколы электронной подписи. Электронная подпись на основе алгоритма RSA

В результате абонент Б получает исходное сообщение и убеждается в том, что его отправил именно абонент А. Данная схема позволяет защититься от нескольких видов возможных нарушений, а именно:

- пользователь А не может отказаться от своего сообщения, если он признает, что секретный ключ известен только ему;
- нарушитель без знания секретного ключа не может ни сформировать, ни сделать осмысленное изменение сообщения, передаваемого по линии связи.

Протоколы электронной подписи. Электронная подпись на основе алгоритма RSA

Данная схема позволяет избежать многих конфликтных ситуаций. Иногда нет необходимости зашифровывать передаваемое сообщение, но нужно его скрепить электронной подписью. В этом случае из приведенного выше протокола исключаются шаги 2 и 4, то есть текст шифруется закрытым ключом отправителя, и полученная последовательность присоединяется к документу. Получатель с помощью открытого ключа отправителя расшифровывает прикрепленную подпись, которая, по сути, является зашифрованным повторением основного сообщения. Если расшифрованная подпись совпадает с основным текстом, значит, подпись верна.

Протоколы электронной подписи.

Существуют и другие варианты применения алгоритма RSA для формирования ЭЦП. Например, можно шифровать (то есть подписывать) открытым ключом не само сообщение, а хеш-код от него.

При создании цифровой подписи по классической схеме отправитель:

- - применяет к исходному сообщению **T** хеш-функцию **h(T)** и получает хеш-образ **r** сообщения;
- - вычисляет цифровую подпись **s** по хеш-образу **r** с использованием своего закрытого ключа;
- - посылает сообщение **T** вместе с цифровой подписью **s** получателю.

Протоколы электронной подписи.

• Получатель, отделив цифровую подпись от сообщения, выполняет следующие действия:

• - применяет к полученному сообщению **T** хеш-функцию **h(T)** и получает хеш-образ **r** сообщения;

• - расшифровывает хеш-образ **r'** из цифровой подписи **s** с использованием открытого ключа отправителя;

• - проверяет соответствие хеш-образов **r** и **r'** и если они совпадают, то отправитель действительно является тем, за кого себя выдает, и сообщение при передаче не подверглось искажению.

Протоколы электронной подписи.

- Как видно из этой схемы, порядок использования ключей обратный тому, который используется при передаче секретных сообщений. Вначале отправитель использует свой закрытый ключ, а затем получатель применяет открытый ключ отправителя.
- Существует несколько схем ЭЦП, которые, как правило, применяются совместно с определенными хеш-функциями.

Протоколы электронной подписи.

Протокол на базе алгоритма RSA

Этап 1. Выработка ключей (выполняет отправитель **A**) - см. лекцию "Шифрование с открытым ключом".

Этап 2. Отправка сообщения и электронной подписи (выполняет отправитель **A**).

Таблица. Отправка сообщения и ЭЦП на базе алгоритма RSA

№ п/п	Описание операции	Пример
1	Вычисление хеш-образа **h** = h(T), где **T** – исходное сообщение, **h(T)** – хеш-функция (для MD5 длина хеш-образа 128 бит).	h = 7
2	Выработка цифровой подписи **s** = h^d mod n, где **d** – закрытый ключ отправителя **A**, **n** – часть открытого ключа отправителя **A**.	s = 7^{29} mod 91 =
3	Отправка получателю **B** исходного сообщения **T** и цифровой подписи **s**.	

Протоколы электронной подписи.

Этап 1. Выработка ключей (выполняет отправитель **A**) - см. лекцию "Шифрование с открытым ключом".

Этап 2. Отправка сообщения и электронной подписи (выполняет отправитель **A**).

Таблица. Отправка сообщения и ЭЦП на базе алгоритма RSA

№ п/п	Описание операции	Пример
1	Вычисление хеш-образа **h** = h(T), где **T** – исходное сообщение, **h(T)** – хеш-функция (для MD5 длина хеш-образа 128 бит).	h = 7
2	Выработка цифровой подписи $\mathbf{s} = h^{d} \bmod n$, где **d** – закрытый ключ отправителя **A**, **n** – часть открытого ключа отправителя **A**.	$s = 7^{29} \bmod 91 = 63$
3	Отправка получателю **B** исходного сообщения **T** и цифровой подписи **s**.	

Протоколы электронной подписи.

Этап 3. Получение сообщения и проверка электронной подписи (выполняет получатель **B**).

Таблица. Получение сообщения и проверка ЭЦП на базе алгоритма RSA

№ п/п	Описание операции	Пример
1	Вычисление хеш-образа по полученному сообщению **h'** = h(T'), где **T'** – полученное сообщение. Если T = T', то должно быть h = h'.	h' =
2	Вычисление хеш-образа из цифровой подписи **h** = s^{e} mod n, где **e** и **n** – открытый ключ отправителя **A**.	h = ^{5} mod =
3	Т.к. h' = h, то получатель **B** делает вывод, что полученное сообщение T' = T и оно действительно отправлено **A**.	

Протоколы электронной подписи.

Этап 3. Получение сообщения и проверка электронной подписи (выполняет получатель **B**).

Таблица. Получение сообщения и проверка ЭЦП на базе алгоритма RSA

№ п/п	Описание операции	Пример
1	Вычисление хеш-образа по полученному сообщению **h'** = h(T'), где **T'** – полученное сообщение. Если T = T', то должно быть h = h'.	h' = 7
2	Вычисление хеш-образа из цифровой подписи **h** = s^e mod n, где **e** и **n** – открытый ключ отправителя **A**.	h = 63^5 mod 91 = 7
3	Т.к. h' = h, то получатель **B** делает вывод, что полученное сообщение T' = T и оно действительно отправлено **A**.	