УДК 003.26:004.056.5

А.В. Жилин, Я.В. Зинченко, А.В. Корнейко

RSA-ПРИМИТИВЫ И ИХ ПРИМЕНЕНИЕ В АЛГОРИТМАХ RSA СО СХЕМАМИ РАНДОМИЗАЦИИ И КОДИРОВАНИЯ ПРИ РЕАЛИЗАЦИИ КРИПТОГРАФИЧЕСКОЙ ЗАЩИТЫ ИНФОРМАЦИИ

В статье приводится анализ существующих RSA-примитивов шифрования и дешифрования и их обозначение в различных документах. Также показан порядок применения этих примитивов со схемами рандомизации и кодирования при построении современных асимметричных криптоалгоритмов.

Ключевые слова: криптографическая защита информации, асимметричные криптоалгоритмы, RSA-примитивы

1. Введение

В настоящее время задача обеспечения защиты информации, циркулирующей в различных информационно-телекоммуникационных системах (ИТС), является неотъемлемой составляющей обеспечения информационной безопасности предприятия, организации, государства. При этом криптографическая защита информации (КЗИ) является одним из важнейших механизмов защиты, так как основана на математически доказанных операциях обеспечения конфиденциальности, целостности и доступности информации.

Основой КЗИ являются криптографические преобразования, реализованные на основе тех или иных криптографических алгоритмов (КА), обеспечивающих соответствующие функции по зашифрованию, расшифрованию или цифровой подписи.

В последние годы определенную нишу в общем комплексе КА получили асимметричные криптоалгоритмы (АКА). Их криптографическая стойкость основывается на трудоемкости вычислительной задачи обращения односторонних функций, лежащих в основе математической модели построения данного АКА.

Рис. 1. Классификация вычислительных задач, лежащих в основе стойкости АКА

Они могут быть использованы для распределения ключей, шифрования передаваемых ключей и формирования общего секретного ключа; обеспечения аутентификации абонентов и ЭЦП; генерации криптографически сильных псевдослучайных последовательностей; генерации параметров ключевых массивов, используемых для создания ключей средств КЗИ и др. [1]. Классификация вычислительных задач, лежащих в основе стойкости АКА, представлена на рис. 1.

Особое место в комплексе АКА, таких как алгоритм Деффи-Хеллмана, Эль-Гамаля и ДСТУ 4145-2005, принадлежит алгоритму RSA. Так как в последние годы было разработано большое количество механизмов атак на протоколы с алгоритмом RSA, то сам алгоритм рекомендован к использованию совместно со схемами рандомизации и кодирования или как составная часть соответствующих гибридных КА [2]. Поэтому целью данной статьи является анализ существующих RSA-примитивов и их применения в алгоритмах RSA со схемами рандомизации и кодирования.

2. Криптоалгоритм RSA и RSA-примитивы

Использование алгоритма RSA (или точнее его соответствующих примитивов) для обеспечения КЗИ в ИТС в настоящее время рекомендовано рядом международных и национальных стандартов, например: ISO/IEC 11166-2:1994, 18033-2:2006 и 9796-2:2010, IEEE Std 1363-2000 и 1363a-2004, PKCS #1, RFC 2437, ANSI X9.44, FIPS 186-3:2009, ITU-T X.509, PEM и др. Кроме этого данный алгоритм рекомендован некоторыми стандартами банковских систем электронных платежей S.W.I.F.T и ANSI X9.31, белорусским стандартом СТБ 34.101.22-2009 и австралийским стандартом управления ключами AS2805.6.5.3 [3].

Алгоритм RSA был разработан в 1978 г. Р. Ривестом (R. Rivest), А. Шамиром (A. Shamir) и Л. Адлеманом (L. Adleman), которые предложили использовать в криптоалгоритме фундаментальную теорему теории чисел, известную как теорема Ферма-Эйлера [4]:

$\beta^{\varphi(N)} \equiv 1 \bmod N$,

где $1 < \beta < N-1$, $(\beta,\ N) = 1$; $\varphi(N)$ – функция Эйлера.

Рассмотрим принцип работы алгоритма RSA при решении задачи обеспечения конфиденциальности сообщения.

Вначале генерируются RSA-ключи следующим образом:

– выбираются два случайных простых числа P и Q заданного размера;

– вычисляется их произведение $N = P \cdot Q$, которое называется криптомодулем RSA;

– вычисляется значение функции Эйлера от числа N: $\varphi(N) = (P-1)\cdot(Q-1)$;

– выбирается число e, взаимно простое с $\varphi(N)$. Обычно в качестве e выбирают простые числа, содержащие небольшое количество единичных бит двоичной записи, например, простые числа Ферма 17, 257, 65537;

– по обобщенному алгоритму Евклида вычисляется число d, которое называется секретной экспонентой и которое мультипликативно обратное числу по модулю $\varphi(N)$, т.е. число, удовлетворяющее условию: $e \cdot d \equiv 1 \bmod \varphi(N)$;

– пара $PU = (e,\ N)$ публикуется в качестве открытого ключа (англ. *public key*), а пара $PR = (d,\ N)$ используется как личный (секретный) ключ (англ. *private key*). Пара $KP = (e,\ d)$ связанных между собой чисел e и d называется ассиметричной ключевой парой (англ. *asymmetric key pair*).

Зашифрование отправителем открытого сообщения, которое представляется целыми числами M из интервала $(1,\ N)$, осуществляется следующим образом:

– выбирается открытый ключ $(e,\ N)$;

– с его помощью шифруется сообщение M: $C = M^e \bmod N$, где C – криптограмма сообщения M.

На приемной стороне криптограмма C расшифровывается получателем с помощью его секретного ключа (d, N) следующим образом (все вычисления в поле N):

$M = C^d \bmod N \equiv \left(M^e\right)^d \equiv M^{e \cdot d} \equiv M^{\varphi(N)k+1} \equiv M^{\varphi(N)} \cdot M \equiv M$.

Рассмотренные выше криптопреобразования, обеспечивающие зашифрование и расшифрование сообщений по классическому алгоритму RSA, называют также RSA-примитивами шифрования и дешифрования и обозначаются как RSAEP и RSADP соответственно [5]. В [6] эти примитивы обозначаются как IFEP-RSA и IFDP-RSA соответственно.

Алгоритм RSA может также использоваться для создания цифровой подписи. Процедура формирования ЭЦП *sign* под сообщением схожа с шифрованием, но в степень закрытого ключа d по вычету n возводится не само сообщение или его части, а дайджест сообщения h. Неотъемлемой частью алгоритмов ЭЦП является хеширование информации. Получателем сообщение M с подписью *sign* будет однозначно аутентифицировано, а авторство сообщения – установлено и доказано по паре ключей $K = (e, d)$.

В [5] определено, что криптопреобразования для формирования ЭЦП и ее проверки с помощью рассмотренного выше классического алгоритма RSA называются RSA-примитивами подписи и проверки подписи и обозначаются как RSASP1 и RSAVP1 соответственно. В [6] эти примитивы обозначаются как IFSP-RSA1 и IFVP-RSA1 соответственно.

3. Требования к выбору параметров алгоритмов RSA

Криптомодуль RSA является произведением двух больших чисел P и Q, и в то же время является частью открытого ключа. Предполагается, что восстановление открытого текста по шифртексту и открытому ключу равносильно разложению числа на два простых сомножителя. При этом существуют методы атаки, предназначенные для вскрытия реализаций криптоалгоритмов на основе RSA-примитивов. Они вскрывают не сам базовый алгоритм, а использующий его протокол [7].

Для нивелирования известных атак при реализации криптоалгоритмов на основе RSA-примитивов и обеспечения стойкости алгоритма RSA выдвигаются следующие требования к выбору параметров этих криптоалгоритмов [8]:

1. Простые числа P и Q выбираются случайно и должны быть большими.
2. Разница $|P - Q|$ должна быть большой.
3. Числа $P \pm 1$ и $Q \pm 1$ должны иметь большой простой делитель.
4. НОД $(P-1, Q-1)$ должен быть небольшим.
5. Секретный ключ d не должен быть слишком маленьким.
6. Числа P и Q должны очень близко совпадать по порядку длины.
7. Числа $(P+1)$ и $(Q+1)$ должны содержать в своем разложении большие простые делители.

4. Применение RSA-примитивов в алгоритмах RSA со схемами рандомизации и кодирования

Одним из вариантов использования криптоалгоритма RSA является так называемый «цифровой конверт». В этой реализации сообщение шифруется одним из симметричных криптоалгоритмов, а затем используется открытый ключ RSA для шифрования секретного ключа симметричного криптоалгоритма, который после шифрования вместе с шифртекстом передается получателю. Даже при выполнении приведенных выше требований, как зашифрованный таким образом секретный симметричный ключ, так и зашифрованное с помощью алгоритма RSA короткое сообщение являются уязвимыми к атакам короткого сообщения. При этом простое добавление фиктивных данных к сообщению затрудняет

работу криптоаналитика, но, приложив дополнительные усилия, он может успешно атаковать зашифрованный текст [2, 7].

По вышеизложенным причинам в последние годы RSA-примитивы рекомендованы к использованию, как правило, не в «чистом» виде, а совместно со схемами рандомизации и кодирования или как составная часть соответствующих гибридных криптоалгоритмов (с использованием как методов асимметричных криптопреобразований, так и симметричных) [2]. Так, например, алгоритм шифрования контент-ключей RSAES-OAEP сочетает криптопреобразования с помощью примитивов RSAEP и RSADP с методом предварительного кодирования EME-OAEP, основанном на алгоритме оптимизированного дополнения асимметричного шифрования OAEP (Optimal Asymmetric Encryption Padding) [5].

На рис. 2 показан общий принцип выполнения процедуры зашифрования (расшифрования) ключей алгоритмом RSA с использованием процедуры OAEP [9]. На рис. 2а: P_1 – замаскированное перед зашифрованием алгоритмом RSA с помощью хеш-функций G и H открытое сообщение (ключ), дополненное последовательностью «000…»; P_2 – сообщение по которому определяют маску G (r). В [6] этот алгоритм назван IFES, где используются примитивы шифрования и дешифрования IFEP-RSA и IFDP-RSA. Близким к RSAES-OAEP по построению является алгоритм RSAES-PKCS1-v1.5, который сочетает примитивы RSAEP и RSADP с методом кодирования EME-PKCS1-v1.5 [5].

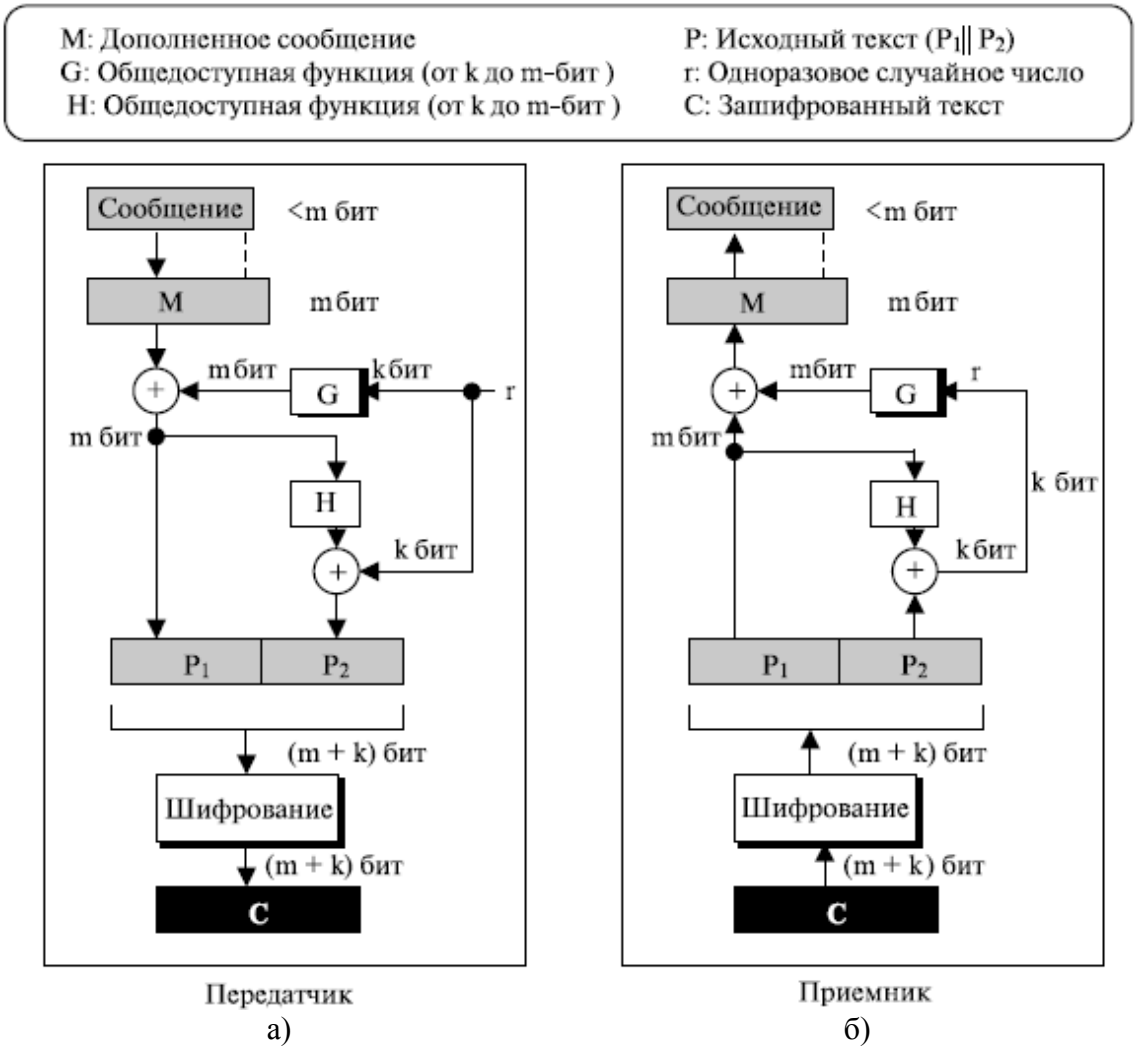
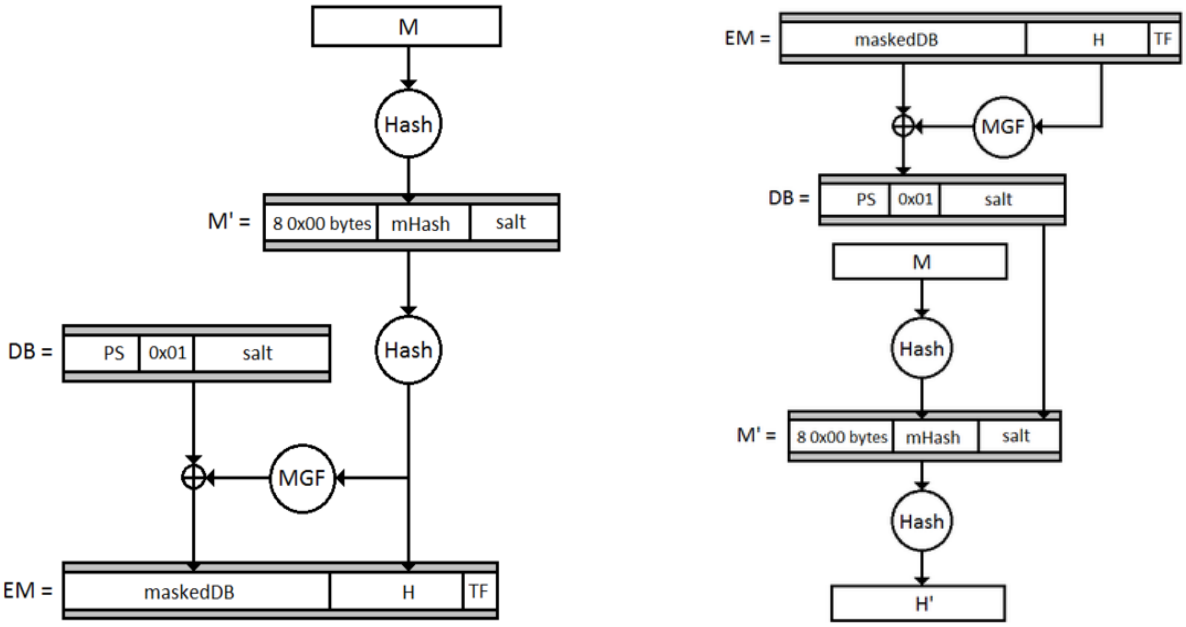


Рис. 2. Принцип выполнения зашифрования (а) и расшифрования (б) сообщений по алгоритму RSA-OAEP

Стандартом ISO/IEC 18033-2:2006 RSA-примитивы рекомендованы к использованию в составе алгоритма RSA-KEM (RSA Key Encapsulation Method), который используется для

создания общего ключа для систем симметричного шифрования и сочетает в себе алгоритм RSA и алгоритм генерации ключа KDF (Key Derivation Function). Алгоритм RSA-KEM рекомендован ISO/IEC 18033-2:2006 для применения в современных протоколах PKCS#1 v1.5, широко используемых в настоящее время в ИТС [10].

Стандартом PKCS #1 v2.1 рекомендован к использованию алгоритм создания ЭЦП с добавлением RSASSA-PSS, который сочетает RSA-примитивы подписи RSASP1 и проверки подписи RSAVP1 с методом рандомизированного кодирования EMSA-PSS [5]. Схема заполнения PSS (Probabilistic Signature Scheme) основана на типовых функциях хеширования (см. рис. 3) [9]. Близким к RSASSA-PSS по построению является алгоритм RSASSA-PKCS1-v1.5, где примитивы RSASP1 и RSAVP1 используются вместе с методом кодирования EMSA-PKCS1-v1.5 [5]. Он совместим с алгоритмом, определенным в [6] как IFSSA, где RSA-примитивы подписи IFSP-RSA1 и проверки подписи IFVP-RSA1 используются вместе с методом кодирования EMSA4.



а) кодирование при подписании б) декодирование при верификации

Рис. 3. Принцип работы схемы PSS при создании (а) и верификации (б) ЭЦП алгоритмом RSASSA-PSS

Обозначения на рисунке:

- M – подписываемое сообщение;
- Hash – хэш-функция, возвращает байтовую строку;
- MGF – mask generation function, преобразует входную байтовую строку в строку заданной длины;
- sLen – длина байтовой строки salt;
- EM – строка M, полученная в результате PSS-кодирования;
- salt – сгенерированная случайная строка;
- mHash – результат применения хэш-функции Hash к сообщению M;
- M’ – строка, состоящая из первых 8 нулевых байт, mHash и salt;
- H – результат применения хэш-функции Hash к сообщению M’;
- PS – сгенерированная строка;
- EM – закодированное сообщение;
- H’ – результат применения хэш-функции Hash к сообщению M’ при проверке подписи.

5. Заключение

Распространение алгоритма RSA как для решения задач шифрования сообщений и передаваемых ключей, обеспечения аутентификации абонентов и ЭЦП, так и для других задач привело к появлению методов атак на протоколы, которые используют этот криптоалгоритм. Поэтому рекомендовано использование алгоритма RSA совместно со схемами рандомизации и кодирования или как составной части соответствующих гибридных криптоалгоритмов. Так, проведенный анализ показал, что RSA-примитивы применяются в алгоритме шифрования контент-ключей RSAES-OAEP вместе с методом предварительного кодирования EME-OAEP и в алгоритме создания ЭЦП с добавлением RSASSA-PSS вместе с методом рандомизированного кодирования EMSA-PSS. Также эти примитивы совместно с алгоритмом генерации ключа KDF используются в алгоритме создания общего ключа для систем симметричного шифрования RSA-KEM. В дальнейшем предполагается провести анализ уязвимости представленных схем рандомизации и кодирования и их влияния на общую криптостойкость рассматриваемых криптоалгоритмов.

ЛИТЕРАТУРА

1. Ростовцев А.Г. Теоретическая криптография / А.Г. Ростовцев, Е.Б. Маховенко. – СПб: АНО НПО «Профессионал», 2004. – 480с.

2. Бернет С. Криптография. Официальное руководство RSA Security / С. Бернет, С. Пэйн. – М.: Бином-Пресс, 2002 г. – 384 с.

3. Жилин А.В., Корнейко А.В., Мохор В.В. Использование RSA алгоритма для обеспечения задач криптографической защиты информации в современных информационно-телекоммуникационных системах // Захист інформації. – 2013. – Том 15. – № 3. – С. 225-230.

4. Adleman L. A method for obtaining digital signatures and public-key cryptosystems / L. Adleman, R.L. Rivest, A. Shamir // Comm. ACM 21. – 1978. – P. 120-126.

5. RSA Cryptography Standard: PKCS #1 v 2.1 – RSA Laboratories, 2002. – P. 62.

6. Standard Specifications for Public Key Cryptography: IEEE Std 1363-2000. – IEEE, 2000.

7. Song Y. Yan. Cryptanalytic attacks on RSA / Song Y. Yan – Springer Science and Business Media, Inc. 2008. – P. 255

8. Коутинхо С. Введение в теорию чисел. Алгоритм RSA / С. Коутинхо. – М.: Постмаркет, 2001. – 328 с.

9. Мао Венбо. Современная криптография: теория и практика: пер. с англ./ Мао Венбо – М. : Изд. дом «Вильямс», 2005. – 768 с.

10 Информационные технологии. Методы и средства обеспечения безопасности. Алгоритмы шифрования. Часть 2. Асимметрические коды: ISO/IEC 18033-2:2006, 2006.

Надійшла: 12.02.2014 р. Рецензент: д.т.н., проф. Барабаш О.В.