УДК 004.056.55

Разработка комбинированного алгоритма шифрования мультимедийных данных в процессе их передачи

Барабошкин Д. А., Бакаева О. А.

Национальный исследовательский
Мордовский государственный университет им. Н. П. Огарёва

Аннотация: В статье представлено описание процесса разработки комбинированного алгоритма шифрования мультимедийных данных, суть которого состоит в применении существующих алгоритмов на разных этапах шифрования.

Ключевые слова: комбинированный алгоритм, шифрование данных, криптостойкость, ключ, хеш, пакет, процесс Handshake, безопасность, конфиденциальность.

Передача информации в современном виде была бы невозможна без существования протоколов передачи данных. В основе протоколов лежат алгоритмы шифрования данных. Существующие алгоритмы: AES, RSA, протокол Диффи-Хеллмана и функция хеширования данных SHA256 – имеют ряд недостатков с точки зрения безопасности передачи и хранения данных пользователя [1, 2]. Поэтому существует необходимость разработки универсального алгоритма шифрования мультимедийных данных.

Комбинированный алгоритм основан на инициировании и передаче RSA ключей, а также ключей Диффи-Хелмана, необходимых для осуществления процесса «рукопожатия» Handshake – установки защищенного соединения между инициатором и респондером.

Процесс Handshake включает в себя три шага. На первом шаге процесса Handshake происходит обмен RSA ключами. Он представлен на рис. 1.

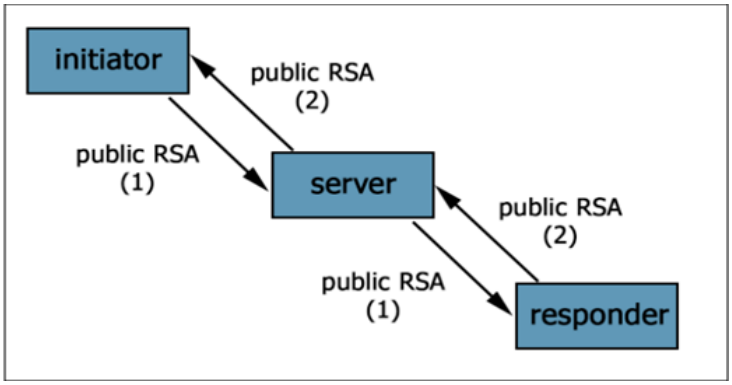

Рис. 1. Обмен RSA ключами между инициатором и респондером

Идентичность проверяется посредством сравнения хеша от присланного public RSA с `sender_id`. Далее, если они не совпадают, Handshake прерывается.

На втором шаге процесса Handshake происходит обмен Диффи-Хелман ключами, представленный на рис. 2.

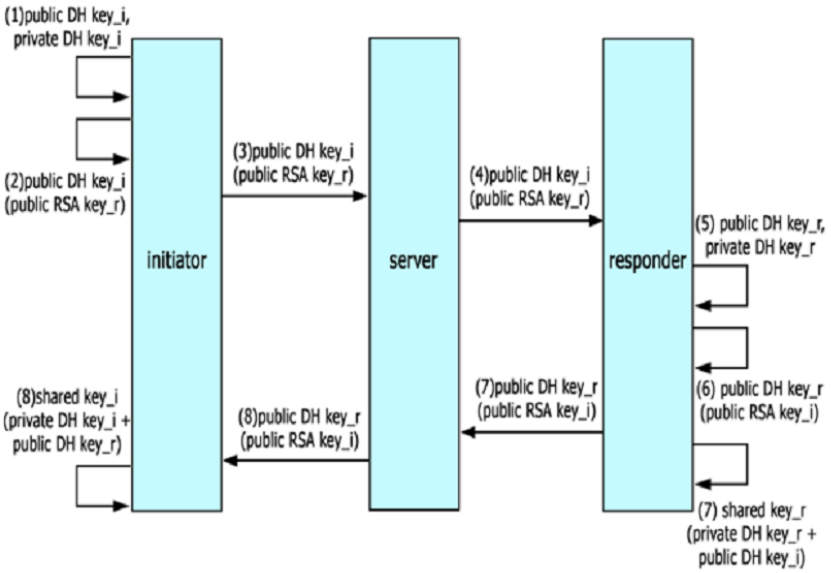

Рис. 2. Алгоритм генерации и передачи ключей Диффи-Хелмана

Алгоритм генерации и обмена Диффи-Хелман ключами представляет собой следующую последовательность действий:

1) на стороне инициатора генерируется пара public и private Диффи-Хелман ключей;

2) public Диффи-Хелман ключ шифруется полученным public RSA собеседника;

3) зашифрованный public Диффи-Хелман ключ отправляется собеседнику;

4) сервер пересылает public Диффи-Хелман ключ собеседнику;

5) на стороне респондера генерируется пара public и private Диффи-Хелман ключей;

6) public Диффи-Хелман ключ шифруется полученным public RSA собеседника;

7) зашифрованный public Диффи-Хелман ключ отправляется собеседнику; далее происходит генерация shared key с помощью private Диффи-Хелман ключа от респондера и public Диффи-Хелман ключа от инициатора;

8) сервер пересылает public Диффи-Хелман ключ собеседнику.

Далее инициатор принимает сообщение сервера и генерирует shared key.

После 7-го шага responder создает «крипто-материал». Это происходит следующим образом.

Берется shared key и к нему добавляются байты: 0, 1, 2.

1 – shared key | 0

2 – shared key | 1

3 – shared key | 2

Полученные три «крипто-материала» хешируются с помощью SHA-256 и от результатов хеша получаются:

1 – `session_id`;

2 – AES key (`encryption_key` для initiator, `decryption_key` для responder);

3 – AES key (`encryption_key` для responder, `decryption_key` для initiator).

После завершения второго шага устанавливается защищенное соединение и на-

чинается третий шаг процесса Handshake, в котором происходит обмен полученными `session_id` между инициатором и респондером.

Полученный `session_id` отправляется собеседнику. Собеседник отправляет свой, полученный `session_id`. Происходит сравнение `session_id` собеседников. Они должны получиться одинаковыми. Тогда соединение будет установлено. Если они различны – соединение обрывается.

Таким образом, в процессе Handshake происходит обмен public RSA ключами между инициатором и респондером, генерация пар public и private DH keys на сторонах клиентов. С помощью пары private DH key и public DH key генерируется shared key, создаются 3 «крипто-материала». Они хешируются с помощью SHA-256 и от результатов хеша получаются пары `encryption_key` (AES key для initiator), `decryption_key` (AES key для responder) и `encryption_key` (AES key для responder), `decryption_key` (AES key для initiator) [3].

В процессе шифрования при отправке сообщения данные шифруются с помощью AES.

Для осуществления процесса установки защищенного соединения были разработаны пакеты, описанные ниже и представленные на рис. 3-4 [4].

`INIT_HANDSHAKE_REQUEST` = 100 (Инициализация Handshake)
key length – 2 байта
key – key length байт

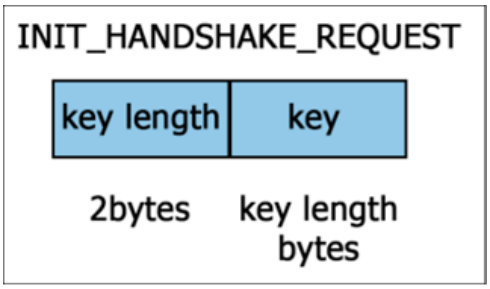

Рис. 3. Структура payload INIT HANDSHAKE REQUEST

`INIT_HANDSHAKE_RESPONSE` = 101
status – 1 байт
0 – none (discard)
1 – success (продолжаем Handshake)
key length – 2 байта
key – key length байт

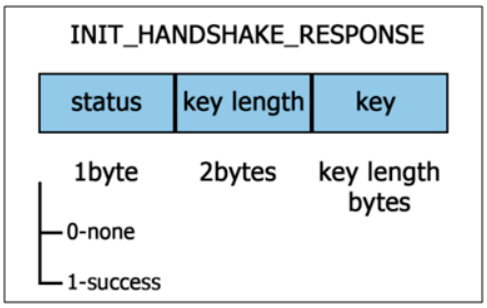

Рис. 4. Структура payload INIT HANDSHAKE RESPONSE

First step – обмен public DH keys, зашифрованными с помощью public RSA ключа

собеседника.

`HANDSHAKE_FIRST_STEP_INITIATOR` = 200

DH key, зашифрованный с помощью public RSA ключа собеседника, полученного в предыдущем шаге.

`HANDSHAKE_FIRST_STEP_RESPONDER` = 201

DH key, зашифрованный с помощью public RSA ключа собеседника, полученного в предыдущем шаге.

Second step – проверка правильности полученных параметров защищенного канала `session_id` и пары полученных shared keys ключей. Они проверяются на совпадение у initiator и responder.

`HANDSHAKE_SECOND_STEP_INITIATOR` = 300

`session_id` зашифрованный своим `encryption_key`.

`HANDSHAKE_SECOND_STEP_RESPONDER` = 301

`session_id` зашифрованный своим `encryption_key`.

При разработке комбинированного алгоритма шифрования данных использовались [1]:

1) криптографический протокол Диффи-Хелмана;

2) алгоритм хеширования SHA-256;

3) симметричный алгоритм блочного шифрования AES с размером ключа 128 бит в режиме шифрования CTR;

4) криптографический алгоритм с открытым ключом RSA.

Получение хеша и проверка целостности получаемого пакета осуществляется с помощью алгоритма SHA-256.

Данный алгоритм позволяет обеспечить максимальный уровень безопасности передачи данных и тем самым сохранить приватность пользователя.

Литература

1. Барабошкин Д.А., Бакаева О.А. Анализ алгоритмов шифрования данных // Сборник научных статей 3-й Всероссийской молодежной научной конференции «ЗА НАМИ БУДУЩЕЕ: взгляд молодых ученых на инновационное развитие общества». Юго-Зап. гос. ун-т, Курск. 2022. Т. 2. С. 449–452.

2. Старусев О.Г. Формальное описание функционирования протоколов передачи данных // Вестник НТУ ХПИ. 2005. №46. – URL: https://cyberleninka.ru/article/n/formalnoe-opisanie-funktsionirovaniya-protokolov-peredachi-dannyh

3. Вотинов М.В. Практикум по архитектуре вычислительных машин, комплексам защиты информации и протоколам передачи данных в компьютерных сетях. Мурманск: МГТУ, 2018. 110 с.

4. Беляев С.В., Совин А.В., Солошенко Е.Б. История развития последовательных протоколов передачи данных // Оборонный комплекс научно-техническому прогрессу России. 2007. №2. С. 55–60. – URL: https://elibrary.ru/item.asp?id=11902688.

MSC 68P25

The development of a combined algorithm for encrypting multimedia data during transmission

D. A. Baraboshkin, O. A. Bakaeva

National Research Mordovia State University

Abstract: The article presents a description of the process of developing a combined algorithm for encrypting multimedia data, the essence of which is to apply existing algorithms at different stages of encryption.

Keywords: combined algorithm, data encryption, cryptographic strength, key, hash, packet, Handshake process, security, privacy.

References

1. D.A. Baraboshkin, O.A. Bakaeva, Analysis of data encryption algorithms, *Collection of scientific papers of the 3rd All-Russian youth scientific conference "FOR WE ARE THE FUTURE: the view of young scientists on the innovative development of society"*, South-West. State University, 2022, Vol. 2, P. 449–452.

2. O.G. Starusev, Formal Description of the Functioning of Data Transmission Protocols, *Newsletter of NTU KhPI*, 2005, №46. URL: https://cyberleninka.ru/article/n/formalnoe-opisanie-funktsionirovaniya-protokolov-peredachi-dannyh

3. M.V., Votinov Workshop on the architecture of computers, information protection systems and data transmission protocols in computer networks, Murmansk, MSTU, 2018, 110 p.

4. S.V. Belyaev, A.V. Sovin, E.B. Soloshenko, History of Serial Data Transmission Protocols Development, *Defense Complex to Scientific and Technical Progress of Russia*, 2007, №2, C. 55-60. URL: https://elibrary.ru/item.asp?id=11902688.