УДК 511

Д. О. Олефиренко, С. И. Алешников
М. В. Алешникова, И. А. Ветров

РАЗРАБОТКА И ИССЛЕДОВАНИЕ СВОЙСТВ ПРОТОКОЛОВ ПОЧТОВОГО ОБМЕНА ВНУТРИ ГИБРИДНОГО ОБЛАКА С ДВУМЯ ПЕРЕШИФРОВАНИЯМИ НА ОСНОВЕ СПАРИВАНИЯ ХЕССА

Рассмотрено спаривание Хесса на эллиптических кривых и построен алгоритм его вычисления. Основной целью статьи является разработка и обзор свойств протокола почтового обмена внутри гибридного облака с двумя перешифрованиями на основе этого спаривания. Отдельным пунктом рассмотрен применяемый в общей схеме алгоритм аутентификации.

In this article we describe Hess pairing on elliptic curves and create its computational algorithm. Main purpose of this article is creation and properties overview of mail-exchange protocol inside hybrid cloud with two re-encryptions, based on this pairing. Also, we describe an authentification algorithm, used in this protocol.

Ключевые слова: эллиптическая кривая, спаривание Хесса, облачное хранилище, аутентификация, протокол.

Keywords: elliptic curve, Hess pairing, cloud storage, authentification, protocol.

Введение

В настоящее время остро встает вопрос о хранении данных. Хранить все нужные данные на жестких дисках становится тяжело из-за постоянно увеличивающегося их объема. Отличным решением этой проблемы стала разработка облачных хранилищ.

Облачные хранилища бывают трех видов: частные, публичные и гибридные. Частные обычно применяются небольшими фирмами и администрируются на месте. Однако их размер, как правило, очень мал, что не позволяет хранить большое количество данных. Но основное их преимущество — высокий уровень безопасности. Публичные, наоборот, имеют большой размер, но администрируются извне. Поэтому их уровень безопасности значительно ниже. Гибридные совмещают в себе плюсы как частных, так и публичных облаков. Информация из частного облака перемещается в публичное в случае нехватки места или же принудительно пользователем. Это позволяет не хранить лишние данные в частном облаке и оставлять только наиболее важные для пользователя.

Вопрос о защите таких облаков сейчас крайне актуален. Популярность и количество пользователей облачных сервисов растут день ото дня. Поэтому разработка системы защиты с применением современных математических средств и является целью настоящей работы.

1. Схема перешифрования

Пусть владелец данных — это пользователь A, а получатель — пользователь B (рис. 1). Оба пользователя получают пары «открытый / закрытый ключ» следующим образом.

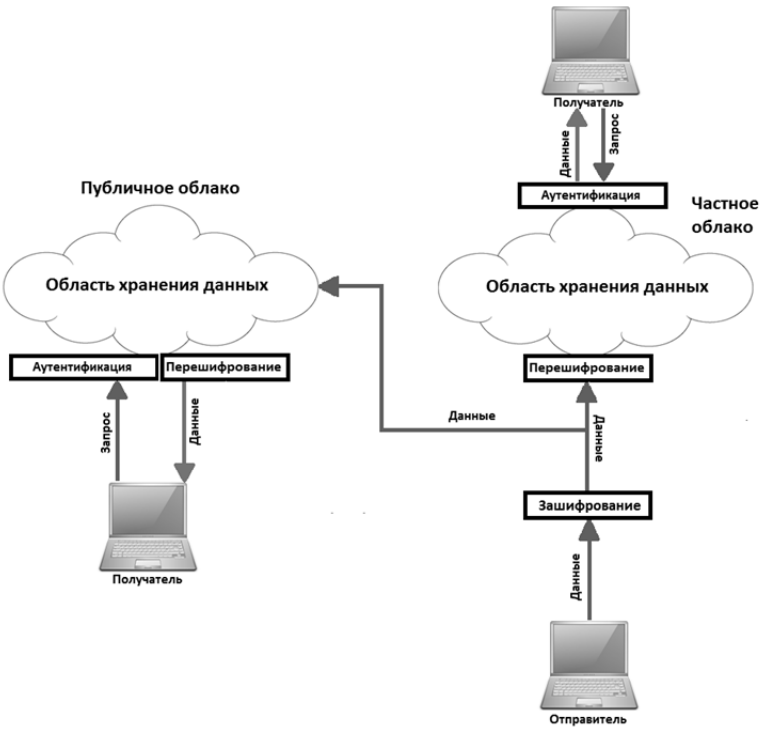

Рис. 1. Общая схема

1.1. Инициализация генератора закрытых ключей

Пусть G_1, G_2 — две циклические группы простого порядка p; $e: G_1 \times G_2 \to G_2$ — билинейное отображение (спаривание Хесса) [5]; G_2 — пространство сообщений; $G_1 \times G_2$ — шифр-пространство. Выбирается случайная образующая ζ группы G_1 (параметр безопасности). Генератор закрытых ключей (ГЗК) случайным образом выбирает целое $s \in F_p^*$ в качестве мастер-ключа и публикует $s * \zeta$ [1].

1.2. Извлечение секретного ключа

Пользователь A с идентификатором ID_A запрашивает у ГЗК частичное извлечение секретного ключа. ГЗК вычисляет $g_A = ID_A, D_A = s * g_A$ и посылает D_A пользователю A. Аналогичная операция проводится и с пользователем B.

1.3. Генерация секретного значения

Пользователь A случайно выбирает целое $x_A \in F_p^*$. Аналогичным образом поступает и пользователь B.

1.4. Генерация секретного ключа

Пользователь A вычисляет секретный ключ:

$$sk_A = x_A * D_A = s * x_A * g_A,$$

где sk_A держится в секрете. Аналогичную операцию проводит и пользователь B. Чтобы достигнуть дешифровального соглашения (позволить другим расшифровать данные, изначально зашифрованные пользователем A), A также выбирает случайное целое t, являющееся дешифровальным соглашением.

1.5. Генерация открытого ключа

Пользователь A вычисляет свой открытый ключ:

$$pk_A = (g_A, s * x_A * \zeta).$$

Публикуется pk_A, и все, кто хочет послать сообщение A, могут использовать этот ключ для шифрования.

1.6. Зашифрование

Чтобы зашифровать сообщение $m \in G_1$, которое может расшифровать только сам владелец, A случайно выбирает целое r и вычисляет

$$c = C_A(m) = (r * \zeta, m * e(g_A, s * x_A * \zeta)^r).$$

Для доверенного расшифрования пользователь A также случайно выбирает целое r и вычисляет

$$C'_A = c' = (pk_B(t * r), r * \zeta, m * e(g_A, s * x_A * \zeta)^r),$$

где $pk_B(t * r) = C_B(t * r)$ — результат шифрования $t * r$ с помощью открытого ключа pk_B пользователя B. В описанном нами ранее сценарии обмена данными m — это КЗД для группы обмена, созданной пользователем A.

1.7. Расшифрование

Чтобы расшифровать $C_A(m) = (u, v)$, используя sk_A, пользователь A вычисляет

$$\frac{v}{e(sk_A, u)} = m * \frac{e(g_A, s * x_A * g)^r}{e(s * x_A * h_A, r * \zeta)} = m.$$

1.8. Генерация прокси-ключа перешифрования

Если пользователь A хочет наделить правами расшифрования пользователя B, A случайно выбирает $x \in G_2$ и вычисляет прокси-ключ перешифрования

$$rk_{A\to B} = \left(-s * x_A * h_A + t * x, C_B(x)\right),$$

который затем отправляет в прокси.

1.9. Прокси-перешифрование

Чтобы перешифровать шифр текст $C'_A(m)$, используя ключ перешифрования $rk_{A\to B}$, прокси вычисляет

$$c'' = m \cdot e(g_A, s * x_A * \zeta)^r * e(-s * x_A * h_A + t * x, r * \zeta) = m * e(t * x, r * \zeta)$$

и затем отправляет $(pk_B(t * r), c'', C_B(x))$ пользователю B.

1.10. Расшифрование перешифрованного

После получения $(pk_B(t * r), c'', C_B(x))$ пользователь B расшифровывает $C_B(x)$, чтобы получить x, и затем получает сообщение

$$\frac{c''}{e(x, t * r * \zeta)} = m.$$

2. Схема аутентификации

Для нашей схемы выбран тип аутентификации по цифровой подписи. В качестве алгоритма цифровой подписи мы используем ECDSA.

2.1. Регистрация

1. Клиент выбирает логин и пароль.
2. На их основе формируется секретный ключ

$$x = H(login + password),$$

где H — хэш-функция.

3. Выбирается поле F_p, где p, соответственно, характеристика поля.
4. Выбирается эллиптическая кривая E.
5. Выбирается простое q, такое, что оно является порядком одной из циклических подгрупп группы точек эллиптической кривой. Если размерность q в битах меньше размерности в битах $H(x)$, то использовать только левые биты $H(x)$. q должно быть, такое, что $x \in [1, q-1]$.
6. Выбирается точка P порядка q кривой $E(F_p)$.
7. Вычисляется $Q = x * P$.
8. Секретный ключ $sk = x$.
9. Логин и открытые параметры (p, q, P, Q) отправляются на сервер и сохраняются в базе данных.

2.2. Аутентификация

1. Клиент вводит логин и пароль.
2. На их основе формируется секретный ключ

$$x = H(login + password),$$

где H — хэш-функция.

3. Клиент получает от сервера случайное число (RNDserver) и генерирует свое случайное число (RNDclient).
4. $m = H(RNDserver + RNDclient)$.
5. Выбирается случайное $k \in [1, q-1]$.
6. Вычисляется $k * P = (x_1, y_1)$, $r = x_1 mod\ q$. Если $r = 0$, выбрать новое k.
7. Вычисляется $k^{-1} mod\ q$ и $s = k^{-1}(m + xr) mod\ q$. Если $s = 0$, выбрать новое k.
8. На сервер отправляются логин, RNDclient и (r, s).
9. Сервер проверяет корректность ЭЦП с помощью открытых параметров клиента, хранящихся в базе данных.

3. Спаривание Хесса

Для схемы перешифрования было выбрано спаривание Хесса на эллиптических кривых [3]. Оно входит в группу оптимальных спариваний, сокращающих количество операций в алгоритме Миллера [4], то есть гарантированно вычисляется быстрее, чем спаривание Вейля и Тейта. Задать спаривание Хесса можно тремя способами: обыкновенное и скрученное на основе спаривания Тейта и обыкновенное на основе спаривания Вейля. Основное преимущество спаривания Хесса — это наличие предвычислений, позволяющих сократить время работы алгоритма Миллера [3]. Сама же функция задается так, что она имеет наименьшую возможную степень.

3.1. Теорема Хесса

Пусть s — примитивный корень степени k из единицы по модулю r^2. $h \in \mathbb{Z}[x]$, такой, что $h(s) \equiv 0\ mod\ r$.

1. Тогда обыкновенное билинейное спаривание Хесса на основе спаривания Тейта есть

$$a_{s,h}: G_2 \times G_1 \to \mu_r,\ (Q, P) \mapsto f_{s,h,Q}(P)^{\frac{q^k-1}{r}}.$$

2. Если $k|\#Aut(E)$, тогда скрученное билинейное спаривание Хесса есть

$$a_{s,h}^{twist}: G_1 \times G_2 \to \mu_r,\ (P, Q) \mapsto f_{s,h,P}(Q)^{\frac{q^k-1}{r}}.$$

3. Если существует $w \in F_q \cap \mu_{\text{НОК}(2,k)}$, тогда билинейное спаривание Хесса на основе спаривания Вейля есть

$$e_{s,h}: G_1 \times G_2 \to \mu_r,\ (P, Q) \mapsto w f_{s,h,P}(Q)/f_{s,h,Q}(P).$$

Спаривания $a_{s,h}$, $a_{s,h}^{twist}$ и $e_{s,h}$ не вырождены тогда и только тогда, когда $h(s) \not\equiv 0 \, mod \, r^2$.

Соотношение со спариваниями Тейта и Вейля выглядят следующим образом:

$$a_{s,h}(Q,P) = t(Q,P)^{h(s)/r},$$
$$a_{s,h}^{twist}(P,Q) = t(P,Q)^{h(s)/r},$$
$$e_{s,h}(P,Q) = e(P,Q)^{h(s)/r}.$$

3.2. Алгоритм вычисления спаривания Хесса

1. Выбрать поле $K = F_q$ и подходящую эллиптическую кривую E. Вычислить $\#E(F_q)$.

2. Среди простых делителей $\#E(F_q)$ выбрать $r \geq 5$. Если таких делителей не нашлось, вернуться к первому пункту.

3. Определить сбалансированную степень $k \geq 2$, $k|(r-1)$ и сбалансированное поле $L = F_{q^k}$. Если необходимые условия не выполняются, вернуться к первому пункту.

4. Определить базис P, Q множества $E\left(F_{q^k}\right)[r]$, удовлетворяющий условиям $\pi(P) = P$ и $\pi(Q) = qQ$. Задать $G_1 =< P >$ и $G_2 =< Q >$, причем $G_1 \cap G_2 = \{\mathcal{O}\}$.

5. Вычислить s, такое, что $s^k \equiv 1 \, mod \, r$.

6. Выбрать $m = \phi(k)$ и построить матрицу M размера $m \times m$.

7. Вычислить кратчайшую линейную комбинацию $w = (w_0, w_1, ..., w_{m-1})$ строк M, используя первый LLL редуцированный базисный элемент, полученный с помощью LLL алгоритма, примененного к строкам M.

8. Вычислить $h = \sum_{i=0}^{m-1} w_i x^i$.

9. При помощи алгоритма Миллера вычислить $f_{s,h,Q}(P)^{(q^k-1)/r}$. Проверить, лежит ли результат в μ_r:

а) вычислить $R_i = s^i Q$, $s_i = \sum_{j=0}^{i} h_j s^j$, $S_i = s_i Q = \sum_{j=0}^{i} h_j R^j$ для $i \geq 0$ и положить $s_{-1} = 0$, $S_{-1} = \mathcal{O}$;

б) $f_{s,h,Q} = \prod_{i=0}^{\deg h} f_{h_i,R_i} \prod_{i=0}^{\deg h} \frac{l_{S_{i-1},h_i R_i}}{v_{S_i}}$,

где $v_{S_i} = X - x_{S_i}$, $l_{S_{i-1},h_i R_i} = \left(Y - y_{S_{i-1}}\right) - \lambda_{S_{i-1},h_i R_i}\left(X - x_{S_{i-1}}\right)$

и

$$\lambda_{S_{i-1},h_i R_i} = \begin{cases} \frac{y_{h_i R_i} - y_{S_{i-1}}}{x_{h_i R_i} - x_{S_{i-1}}}, S_{i-1} \neq h_i R_i \\ \frac{3x_{S_{i-1}}^2 + 2a_2 x_{S_{i-1}} + a_4}{2y_{S_{i-1}} + a_1 x_{S_{i-1}} + a_3}, S_{i-1} = h_i R_i \end{cases}.$$

R_i вычисляется отдельно, а каждое f_{h_i,R_i} — по алгоритму Миллера;

в) полученный результат возвести в степень $(q^k - 1)/r$.

Заключение

В данной статье мы рассмотрели протокол почтового обмена внутри гибридного облака с двумя перешифрованиями на основе спаривания Хесса на эллиптических кривых. Был подробно описан процесс перешифрования и предшествующий ему алгоритм аутентификации.

Данный протокол может быть применен для решения военно-технических задач. Например, для реализации почтового обмена в тыловых учреждениях. Руководство может отправлять письма в такое гибридное облако, и только определенная группа подчиненных, назначенная руководством, будет способна извлечь и прочитать письмо. Это позволит избежать утечки данных и реализовать разделение получателей на группы. Двойное перешифрование данных, находящихся в публичном облаке, гарантирует нераскрытие содержимого в случае кражи из облака. Также этот протокол можно применять для решения и других военно-технических задач. Наличие публичного облака позволит упростить взаимодействие военных организаций с гражданскими и сделает общение более гибким.

Описанный протокол обеспечивает высокий уровень безопасности при допустимой скорости вычислений. Главным его преимуществом является использование такого математического аппарата, как эллиптические кривые и спаривания на них. Поэтому данный протокол полностью отвечает поставленной нами актуальной задаче.

Список литературы

1. *Алешников С.И., Алешникова М.В., Горбачёв А.А.* Протокол доверенного шифрования на основе модифицированного алгоритма вычисления спаривания Вейля на алгебраических кривых для облачных вычислений // Информационные технологии. 2013. №9. С. 36—39.

2. *Enge A.* Bilinear Pairings on Elliptic Curves. 2013. HAL Id: hal-00767404.

3. *Hess F.* Pairing Lattices // S.D. Galbraith, K. Paterson (eds.) Pairing-Based Cryptography. Berlin, 2008. P. 18—38.

4. *Miller V.* The Weil Pairing, and its Efficient Calculation // Journal of Cryptology. 2004. №17. С. 235—261.

5. *Wu X., Xu L., Zhang X.* CL-PRE: a Certificateless Proxy Re-Encryption Scheme for Secure Data Sharing with Public Cloud // Proceedings of the 7th ACM Symposium on Information, Computer and Communications Security. 2012.

Об авторах

Денис Олегович Олефиренко — асп., Балтийский федеральный университет им. И. Канта, Россия.

E-mail: denis_cooler_1@mail.ru

Сергей Иванович Алешников — канд. техн. наук, доц., Балтийский федеральный университет им. И. Канта, Россия.

E-mail: elliptec@mail.ru

Марина Валерьевна Алешникова — ст. преп., Балтийский федеральный университет им. И. Канта, Россия.
E-mail: aleshnikova_m_v@mail.ru

Игорь Анатольевич Ветров — канд. техн. наук, доц., Балтийский федеральный университет им. И. Канта, Россия.
E-mail: vetrov.gosha209@yandex.ru

The authors

Denis O. Olefirenko, PhD Student, I. Kant Baltic Federal University, Russia.
E-mail: denis_cooler_1@mail.ru

Dr Sergey I. Aleshnikov, Associate Professor, I. Kant Baltic Federal University, Russia.
E-mail: elliptec@mail.ru

Marina V. Aleshnikova, Assistant Professor, I. Kant Baltic Federal University, Russia.
E-mail: aleshnikova_m_v@mail.ru

Dr Igor A. Vetrov, Associate Professor, I. Kant Baltic Federal University, Russia.
E-mail: vetrov.gosha209@yandex.ru