УДК 51:003.26

ШИФРОВАНИЕ ТЕКСТОВЫХ ДАННЫХ АЛГОРИТМОМ ELLIPTIC CURVE CRYPTOGRAPHY

М.А. Бурнин
Научный руководитель: доцент, к.ф.-м.н. О.Л. Крицкий
Национальный исследовательский Томский политехнический университет,
Россия, г. Томск, пр. Ленина, 30, 634050
E-mail: mab54@tpu.ru

ENCRYPTION OF TEXT DATA BY ELLIPTIC CURVE CRYPTOGRAPHY ALGORITHM

M.A. Burnin
Scientific Supervisor: Assoc. Prof., PhD. O.L. Kritsky
Tomsk Polytechnic University, Russia, Tomsk, Lenin strt., 30, 634050
E-mail: mab54@tpu.ru

Abstract. *In this work, you have constructed a program that allows us to encrypt a message based on some initial code. We get this code according to the algorithm based on elliptic curves. It is agreed between the two parties and cannot be established based on the data that we transmited through an open communication channel.*

Введение. В настоящее время данные которые вы оставляете в интернете тщательно собирают, и речь идёт не только о cookie файлах, зачастую личная переписка оказывается под угрозой вскрытия, если вы передаёте какую-то важную информацию, то можете перестраховаться и зашифровать её[1]. Данная работа основана на эллиптических кривых, которые ограничены вещественными полями вводимой информации. Рассмотрим понятие эллиптических кривых [1].

Теоретические основы. Кривые вида $y^2 = x^3 + ax + b$, где $4a^3 + 27b^2 \neq 0$ называются эллиптическими. Можно определить абелеву группу для эллиптических кривых [1, 2]:

- элементы группы — это точки эллиптической кривой;
- единичный элемент — это бесконечно удалённая точка 0;
- обратная величина точки P — это точка, симметричная относительно оси x;
- сумма трёх ненулевых точек P, Q, R, лежащих на одной прямой равна $P + Q + R = 0$.

Есть возможность определить скалярное умножение: $nP = P + P + \ldots + P$ (n раз). Которое можно быстро посчитать по алгоритму удвоения-сложения. Далее будем работать в поле целых чисел по модулю p, где p — простое число. Можно определить деление по модулю p: $\frac{x}{y} = x \cdot y^{-1}$, где $\left(y^{-1} \cdot y\right) \bmod p = 1$.

Алгебраическая сумма в ограниченном поле:

$$x_R = (m^2 - x_P - x_Q) \bmod p,\ y_R = \left(y_P + m(x_R - x_P)\right) \bmod p = \left(y_Q + m(x_R - x_Q)\right) \bmod p$$

[1] Материалы исследования автора [Электронный ресурс]. Доступ свободный. Информационная Система: URL: https://colab.research.google.com/drive/10tu1mATEOPA87Lh35uGcMsGxpLQVOhnh

Если $P \neq Q$, то наклон m принимает форму: $m = (y_p - y_Q)(x_p - x_Q)^{-1} \bmod p$

Если $P = Q$, то $m = (3x_p^2 + a)(2y_p)^{-1} \bmod p$

Введём дополнительные понятия и обозначения:

• Количество точек N в группе называется порядком группы.

• Порядок подгруппы порождённой точкой P это минимальное n такое, что $nP = 0$ (n – простое число, делитель N).

• Кофактор $h = N/n$ (всегда целое).

Для корректной работы алгоритма осталось выбрать базовую точку G:

1. Вычисляем порядок эллиптической кривой N при помощи алгоритма Шуфа.
2. Выбираем порядок n подгруппы. Чтобы алгоритм сработал, число должно быть простым и быть делителем N.
3. Вычисляем кофактор $h = N/n$.
4. Выбираем случайную точку P.
5. Вычисляем $G = hP$.
6. Если число G равно 0, то вернёмся к шагу 4. В противном случае мы нашли генератор подгруппы с порядком n и кофактором h.

Алгоритм построения эллиптической кривой Даффи – Хеллмана (ECDH). Данный алгоритм решает следующую проблему: две стороны (обычно Алиса и Боб) хотят безопасно обмениваться информацией, чтобы третья сторона (посредник, Man In the Middle) мог перехватывать её, но не мог расшифровать.

Сначала Алиса и Боб генерируют собственные закрытые и открытые ключи. У Алисы есть закрытый ключ d_A и открытый ключ $H_A = d_A G$, у Боба есть ключи d_B и $H_B = d_B G$. Затем они обмениваются открытыми ключами H_A и H_B по незащищённому каналу. Посредник не может определить закрытые ключи d_A и d_B, не решив «сложнейшую» задачу дискретного логарифмирования. После чего Алиса и Боб могут спокойно вычислить S:

$$S = d_A H_B = d_A(d_B G) = d_B(d_A G) = d_B H_A$$

Процесс работы программы. Обе стороны знают общий секрет, теперь этот секрет может послужить ключом в алгоритме шифрования текстовых данных. И это можно сделать следующим о

б Теперь это число можно использовать в различных алгоритмах шифрования. Но оно специально было сделано не больше заранее заданного алфавита, чтобы:

```
for letter in message:
    position = alfavit.find(letter)
    if variant == "Зашифровать":
      new_position = (position + key)%leng
    elif variant == "Расшифровать":
      new_position = (position - key)%leng
    if letter in alfavit:
       vald = vald + alfavit[new_position]
    else:
       vald = vald + letter //Добавляем символ к строке
```

s

e

c

r

Таким образом мы просто сдвигаем символ в сообщении на значение длины ключа в зависимости от того хотим мы сообщение зашифровать или расшифровать.

Чтобы показать связь теории с практикой, необходимо привести функцию скалярного умножения, в которой личному ключу соответствует переменная k, а публичному переменная point, умножение происходит по алгоритму удвоения-сложения:

```
def scalar_mult(k, point):
    assert is_on_curve(point)
    if k % curve.n == 0 or point is None:
        return None
    if k < 0:
# k * point = -k * (-point)
        return scalar_mult(-k, point_neg(point))
    result = None
    addend = point
    while k:
        if k & 1:
            result = point_add(result, addend)
        addend = point_add(addend, addend)
        k >>= 1
    assert is_on_curve(result)
    return result
```

Переменная *addend* удваивается после каждой итерации и прибавляется к переменной *result*, это происходит только в том случае, если оператор побитового «И» (обозначается «&») возвращает истину. Функция *is_on_curve(point)* проверяет принадлежность точки к кривой, что помогает определить часто возникающую при работе ошибку, то есть неверно указанные данные

Заключение. Рассмотрены положения алгоритма согласования ключей Elliptic Curve Diffie-Hellman по незащищенному каналу связи. Создана программа шифрования текстовых сообщений, она имеет удобный графический интерфейс. Программа позволяет зашифровать/дешифровать сообщение. Время работы программы для шифрования сообщения 256 символов не превосходит 0,007 сек. По сравнению с существующими аналогами моя программа шифрования проще в использовании и интуитивно понятнее.

СПИСОК ЛИТЕРАТУРЫ

1. Доступно о криптографии на эллиптических кривых [Электронный ресурс]. – Режим доступа: – http://habr.com/ru/post/335906/ (28.01.2023)
2. Болотов А.А. Элементарное введение в эллиптическую криптографию: Алгебраические и алгоритмические основы. – М.: КомКнига, 2006. – 328 с.